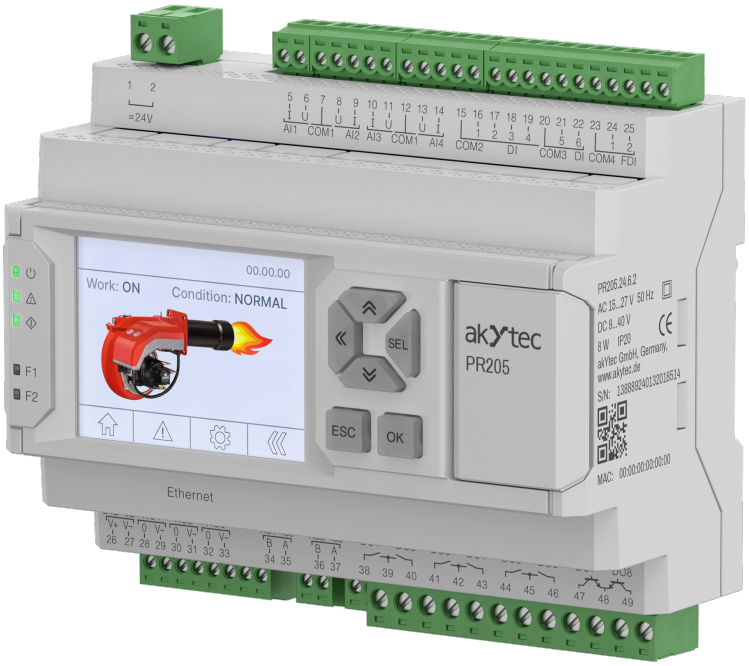
EN




# PR205

## Programmable relay

## User guide

PR205_3-EN-148069-1.5

## Contents

# 1 Introduction

This manual describes functions, configuration, operating instructions, programming and troubleshooting of the PR205 multifunctional programmable relay (hereinafter referred to as PR205, device, or relay).

## 1.1 Terms and abbreviations

- **ALP** – programming software akYtec ALP for programming PR series relays, based on Function Block Diagram programming language (FBD)
- **ADC** – analog-digital converter
- **DAC** – digital-analog converter
- **DNS** (Domain Name System) – decentralized naming system for computers, services, or other resources connected to the Internet or a private network. It translates domain names to the numerical IP addresses.
- **EMI** – electromagnetic interference
- **Modbus** – application layer messaging protocol for client/server communication between devices connected on different types of buses or networks, originally published by Modicon (now Schneider Electric), currently supported by an independent organization *Modbus-IDA*
- **Modbus TCP** – Modbus protocol, adapted to transfer information over TCP
- **NTC** – negative temperature coefficient sensor
- **Project** – user application created in ALP software that also includes the device configuration
- **PTC** – positive temperature coefficient sensors
- **PWM** – pulse-width modulation
- **RAM** – random access memory, volatile part of the device memory
- **Retain memory** – non-volatile device memory for retain variables
- **Retain variable** – type of variable that keeps its value after device restart (power off/on cycle)
- **ROM** – read-only memory, non-volatile part of the device memory
- **RTC** – real time clock
- **RTD** – resistance temperature detectors
- **UTC** (Coordinated Universal Time) – world-wide primary time standard

## 1.2 Symbols and key words

**WARNING**
***WARNING indicates a potentially dangerous situation that could result in death or serious injuries.***

**CAUTION**
***CAUTION indicates a potentially dangerous situation that could result in minor injuries.***

**NOTICE**
***NOTICE indicates a potentially dangerous situation that could result in damage to property.***

**NOTE**
***NOTE indicates helpful tips and recommendations, as well as information for efficient and trouble-free operation.***

### 1.3 Intended use

The device has been designed and built solely for the intended use described here, and may only be used accordingly. The technical specifications contained in this document must be observed.
The device may be operated only in properly installed condition.

**Improper use**

Any other use is considered improper. Especially to note:

– The device may not be used for medical applications.
– The device may not be used in explosive environment.
– The device may not be used in atmosphere in which there are chemically active substances.

### 1.4 Safety requirements

**WARNING**
***All electrical connections must be performed only by a qualified electrician.***
***The device terminals may be under a dangerous voltage. Cut off all power lines before working on the device.***
***Switch on the power supply only after completing all work on the device.***

**WARNING**
***Ensure the mains voltage matches the voltage marked on the nameplate.***
***Ensure the device is provided with its own power supply line and electric fuse.***
***Do not feed any external devices from the power contacts of the device.***

**NOTICE**
***Supply voltage may not exceed 30 V. Higher voltage can damage the device.***
***If the supply voltage is lower than 9 V DC, the device cannot operate properly but will not be damaged.***

**NOTICE**
***If the device is brought from a cold to a warm environment, condensation may form inside the device. To avoid damage to the device, keep the device in the warm environment for at least 1 hour before powering on.***

# 2 Overview

The programmable relay PR205 is a small controller, developed for automated control systems in industry, agriculture, building technology and household applications.
User program is created as a function plan with the ALP programming software.

## 2.1 Basic features

PR205 has the following modifications:

– PR205.24.1.2
– PR205.24.5.2

The basic features of both modifications are described below:

| PR205.24.1.2 | PR205.24.5.2 |
|---|---|
| – 8 digital inputs (including 2 fast digital inputs up to 100 kHz)<br>– 4 analog inputs (can be used as digital)<br>– 8 relay outputs<br>– 3 analog outputs (4-20 mA, 0-10 V)<br>– IPS LCD<br>– 2 programmable LEDs F1, F2<br>– front panel buttons for View and Edit modes<br>– Master or Slave in Modbus network over RS485 interface<br>– Master or Slave in Modbus network over Ethernet interface<br>– real-time clock<br>– data logging and archiving<br>– possibility to connect up to 2 PRM series extension modules<br>– configuration and programming with ALP software (free)<br>– DIN rail mounting<br>– 3-level stepped form for installation in switchboards<br>– quick and easy device replacement with plug-in terminal blocks | – 8 digital inputs (including 2 fast digital inputs up to 100 kHz)<br>– 4 analog inputs (can be used as digital)<br>– 6 relay outputs<br>– 2 outputs of transistor type<br>– 3 analog outputs (4-20 mA, 0-10 V)<br>– IPS LCD<br>– 2 programmable LEDs F1, F2<br>– front panel buttons for View and Edit modes<br>– Master or Slave in Modbus network over RS485 interface<br>– Master or Slave in Modbus network over Ethernet interface<br>– real-time clock<br>– data logging and archiving<br>– possibility to connect up to 2 PRM series extension modules<br>– configuration and programming with ALP software (free)<br>– DIN rail mounting<br>– 3-level stepped form for installation in switchboards<br>– quick and easy device replacement with plug-in terminal blocks |

The number of the PR205 I/O points can be increased by using appropriate PRM extension modules:

– PRM-X.1 — for digital inputs/outputs extension
– PRM-X.2 — for analog inputs/digital outputs extension
– PRM-X.3 — for analog inputs/outputs extension

Please refer to *Section 7.10* for details as to PRM modules connection.

### 2.2 Design

The PR205 front panel is shown in the Figure.

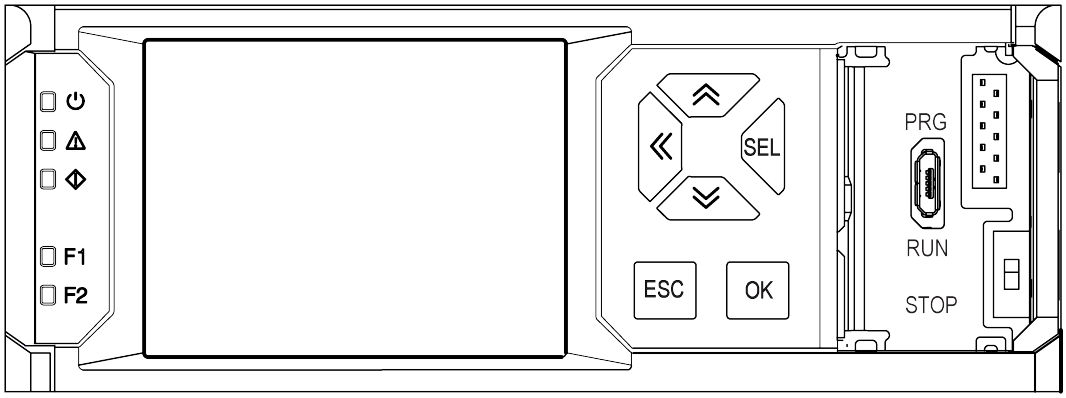



*Fig. 2.1 Front panel (the cover on the right is open)*

– color graphic LCD
– six buttons
– five LEDs
– cover

Under the cover:

– extension module connector
– Run/Stop switch
– microUSB programming connector

The RTC battery slot and the service button are under the PR205 front panel as shown in the Figure.

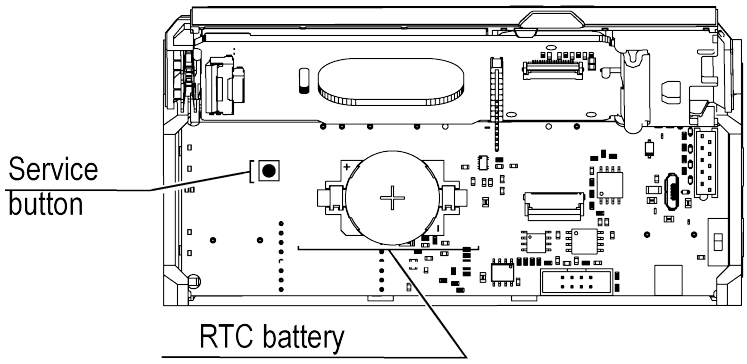
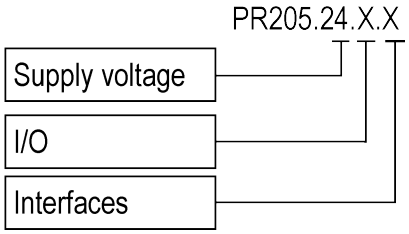



*Fig. 2.2 Under the front panel*

For detailed description of the PR205 indication, controls and interfaces, see *Section 7.1*.

### 2.3 Ordering key

Relay PR205 can be ordered in various designs depending on the number and type of inputs, outputs and interfaces:




**Supply voltage**
24 – 24 (20…36) V DC

**I/O**
1 – 6 DI, 2 FDI, 4 DI/AI, 8 DO, 3 AO (4-20 mA)
5 – 6 DI, 2 FDI, 4 DI/AI, 6 DO (relay), 2 KT (NPN transistor)

**Interfaces**
0 – 1 × Ethernet
2 – 2 × RS485, 1 × Ethernet

# 3 Specifications

## 3.1 Specification tables

*Table 3.1 Power supply specifications*

| Parameter | Value |
|---|---|
| Power supply | 24 (20…36) V DC |
| Power consumption, max. | 8 W |
| Galvanic isolation (between the power input and other circuitry parts) | 510 V |
| Reverse polarity protection | yes |

*Table 3.2 General specifications*

| Parameter | Value |
|---|---|
| **Computing capabilities, extension modules and auxiliaries** | |
| Minimum program execution cycle (depends on user's program complexity) | 1 ms |
| Network variables' memory size:<br>Slave mode<br>Master mode | <br>2040 Bytes<br>384 Bytes |
| Extension modules | up to 2 PRM |
| Built-in real-time clock (RTC) | yes |
| RTC accuracy (at +25 °C) | ± 3 s/day |
| RTC backup battery life | 5 years |
| Backup battery | CR2032 |
| **Flash memory (logging)** | |
| Number of program/erase cycles | 100 000 |
| Maximum log file size, max. | 2048 Bytes |
| Maximum number of log files | 50 |
| Minimum logging interval | 10 s |
| **Programming and configuration** | |
| Programming and configuration interfaces | USB, Ethernet, RS485 |
| Programming software | akYtec ALP |
| Configuration software | akYtec ToolPro |
| Retain memory size | 1024 Bytes |
| ROM | 64 kB |
| RAM | 320 kB |
| User program footprint | 128 kB |
| **Communication interfaces** | |
| Types of network interfaces | RS485 and Ethernet |
| **Ethernet** | |
| Number of Ethernet ports | 1 |
| Communication protocol | Modbus TCP |
| Operation modes | Master / Slave |
| Baud rate | 10/100 Mbit/s |
| Galvanic isolation (between Ethernet port and other circuitry parts) | 510 V |
| **RS485** | |
| Number of RS485 ports | 2 |
| Communication protocol | Modbus-RTU, Modbus ASCII |
| Operation modes | Master / Slave |
| Baud rate | 9600, 14400, 19200,<br>38400, 57600, 115200 bit/s |
| Galvanic isolation (between RS485 ports and other circuitry parts) | 1500 V |

| Parameter | Value |
|---|---|
| **Cloud service** | |
| akYtec Cloud | supported |
| Cloud interface | Ethernet |
| **General** | |
| Mounting | DIN-rail (35 mm) |
| Dimensions (with terminal blocks) | 123 × 96 × 62 mm |
| IP code | IP20 |
| Weight, max. | 0.6 kg |
| Average service life | 8 years |

*Table 3.3 Indication and controls*

| Parameter | Value |
|---|---|
| **Display** | |
| Display type (matrix) | graphic IPS LCD |
| Backlight | LED |
| Colour gamut | 65535 |
| Size | 2.4“ |
| Work area | 49 × 36.7 mm |
| Resolution | 320 × 240 px |
| Brightness | 250 cd/m$^2$ |
| Contrast ratio | 800:1 |
| Top/Right/Bottom/Left Viewing Angle | 80/80/80/80° |
| Backlight MTBF, min | 50 000 hours at 25 °C |
| Backlight control | Yes* |
| Supported languages | English, German |
| **LEDs** | |
| LED description | see *Section 7.1* |
| **Buttons** | |
| Description of function buttons | see *Section 7.1* |

**NOTE**
* Setting a timeout (see *Section 4.10*). Backlight is switched on by pressing any button.

*Table 3.4 Digital inputs*

| Parameter | Value |
|---|---|
| HIGH level | 8.5…40 V / 2…15 mA |
| LOW level | -3…+5 V / 0…15 mA |
| Pulse length, min. | 5 ms |
| Response time, max. | 30 ms |
| Pulse frequency, max. | 150 Hz |
| Galvanic isolation | in groups of 4 |
| Galvanic isolation (between each input group and other circuitry parts) | 510 V |
| Reverse polarity protection | yes |

*Table 3.5 Fast digital inputs*

| Parameter | Value |
|---|---|
| HIGH level | 15…30 V / 2…15 mA |
| LOW level | -3…+5 V / 0…15 mA |
| Pulse length, min. | 5 µs |
| Pulse frequency, max. | 100 kHz |
| Galvanic isolation | in groups |
| Galvanic isolation (between the input group and other circuitry parts) | 510 V |
| Reverse polarity protection | yes |

*Table 3.6 Digital / Analog inputs*

| Parameter | | Value |
|---|---|---|
| Input voltage range | | 0...30 V DC |
| ADC resolution | | 12 bit |
| Sampling time, max. | | 1 ms |
| Galvanic isolation | | no |
| **Analog mode 1 (Linear input)** | | |
| Input signal types | Voltage (unipolar) | 0...10 V |
| | Current (unipolar) | 4...20 mA |
| Input resistance for 0-10 V input | | 10 kΩ |
| Basic error | | ±0.5 % |
| Temperature influence | | ±0.5 % of basic error per each 10 °C |
| Input resistance ( at measuring voltage signal 0...10 V ), min. | | 10 kΩ |
| **Analog mode 2 (Temperature sensors)** | | |
| Input signal range | | 0...300 kΩ |
| Types of temperature sensors supported | | see *Table 3.7* |
| Least significant bit value, max. | | 1 °C |

| Parameter | | Value |
|---|---|---|
| Basic error | for the resistance range 0...150 kΩ | ±1.0 % |
| | for the resistance range 150...300 kΩ | ±2.0 % |
| | for RTD and thermistors (NTC and PTC) | ±1.5 % |
| Temperature influence | | ±0.5 % of basic error per each 10 °C |
| **Digital mode** | | |
| HIGH/LOW threshold (adjustable in ALP) | | 1…8 V |
| LOW/HIGH threshold (adjustable in ALP) | | 2…9 V |
| Input current | | 2...15 mA |
| Pulse length, min. | | 15 ms |
| Signal frequency, max. | | 40 Hz |

**NOTE**
***Interferences can increase the value of basic error.***

*Table 3.7 Sensors (analog mode 2)*

| Sensor | Measurement range |
|---|---|
| **RTD** | |
| Pt 500 (α = 0.00385 °C⁻¹) * | –200…+850 °C |
| 500P (α = 0.00391 °C⁻¹) | –200…+850 °C |
| Cu 500 (α = 0.00426 °C⁻¹) | –50…+200 °C |
| 500M (α = 0.00428 °C⁻¹) | –180…+200 °C |
| Ni500 (α = 0.00617 °C⁻¹) | –60…+180 °C |
| Cu 1000 (α = 0.00426 °C⁻¹) | –50…+200 °C |
| 1000M (α = 0.00428 °C⁻¹) | –180…+200 °C |
| Pt 1000 (α = 0.00385 °C⁻¹) | –200…+850 °C |
| 1000P (α = 0.00391 °C⁻¹) | –200…+850 °C |
| Ni 1000 (α = 0.00617 °C⁻¹) | –60…+180 °C |
| **Thermistors (NTC)** | |
| B57861S series, 2 kΩ, $B_{25/100}$ = 3560K | –55…+100 °C |
| B57861S series, 3 kΩ, $B_{25/100}$ = 3988K | –55…+145 °C |
| B57861S series, 5 kΩ, $B_{25/100}$ = 3988K | –35…+145 °C |
| B57861S series, 10 kΩ, $B_{25/100}$ = 3988K | –35…+155 °C |
| B57861S series, 30 kΩ, $B_{25/100}$ = 3964K | –20…+155 °C |
| B57861S series, 50 kΩ, $B_{25/100}$ = 3760K | –10…+155 °C |
| NTC 3435, 10 kΩ | –40…+105 °C |
| NTC 3977, 10 kΩ | –40…+125 °C |
| **Thermistors (PTC)** | |
| KTY82-110 | –55…+150 °C |
| KTY82-120 | |
| KTY82-121 | |
| KTY82-122 | |
| KTY82-150 | |
| KTY82-151 | |
| KTY82-152 | |

**NOTE**
***Temperature coefficient of resistance (α) is determined by the formula:***
$\alpha = \frac{R_{100} - R_0}{R_0 \cdot 100\ °C}$***, where $R_{100}$, $R_0$ are RTD performance curve resistance values at 100 °C and 0 °C correspondingly. The coefficient value is rounded to the fifth significant figure.***

*Table 3.8 Digital outputs*

| Parameter | | Value |
|---|---|---|
| Type of digital outputs | | relay (NO) |
| Galvanic isolation (between outputs and other circuitry parts) | | 2300 V |
| Maximum switching load voltage | AC | 250 V (resistive load) |
| | DC | 30 V (resistive load) |
| Maximum load current | AC | 5 A (at maximum load voltage 250 V AC and cos(φ) > 0,95) |
| | DC | 3 A (at maximum load voltage 30 V DC) |
| Minimum load current at 5 V DC | | 10 mA |
| Service life, electrical | at 250 V AC, 5 A | 200 000 switching cycles |
| | at 250 V AC, 7 A | 50 000 switching cycles |
| | at 30 V DC, 3 A (resistive load) | 100 000 switching cycles |

*Table 3.9 Transistor outputs*

| Parameter | Value |
|---|---|
| Output type | NPN transistor |
| Maximum switching load voltage | 40 V DC |
| Maximum load current | 0.5 A |
| Overcurrent and short circuit protection | Yes |
| Overheating protection | Yes |
| Galvanic isolation | No |

*Table 3.10 Analog outputs*

| Parameter | | Value |
|---|---|---|
| DAC resolution | | 12 bit |
| Output signal | Current | 4...20 mA |
| | Voltage | 0...10 V |
| External supply voltage | | 15…30 V |
| Load resistance | $R_I$ (4...20 mA), max. | 600 Ω |
| | $R_U$ (0...10 V), min. | 600 Ω |
| Basic error, max. | | ±0.5% |
| Temperature influence | | ±0.5 % of basic error per each 10 °C |
| Galvanic isolation | | No |

### 3.2 Operating conditions

The device is designed for natural convection cooling that should be taken into account when choosing the installation site.
The following environmental conditions must be observed:

– clean, dry and controlled environment, low dust level
– closed non-hazardous areas, free of corrosive or flammable gases

*Table 3.11 Environmental conditions*

| Condition | Permissible range |
|---|---|
| Ambient temperature | -40…+55 °C |
| Relative humidity | up to 80 % (at +35 °C, non-condencing) |
| Transportation and storage temperature | -25…+55 °C |
| Transportation and storage relative humidity | up to 80 % |
| Altitude | up to 2000 m above sea level |
| EMC immunity | conforms to IEC 61000-6-2 |
| EMC emission | conforms to IEC 61000-6-4 |

# 4 Configuration and programming

## 4.1 Putting into operation

It is necessary to observe safety measures (see *Section 1.3*) when putting the device into operation. It is recommended to configure and program the device prior to installation and wiring. Configuration and programming are carried out with ***ALP*** after creating a user project.
The software can be downloaded from our homepage *www.akYtec.de*.
Follow the steps below to put the device into operation:

1. Connect the microUSB programming connector of the device to PC over a USB-to-microUSB connection cable (not included). Please refer to *Fig. 2.2* for location of the microUSB programming connector on the device.
2. Unplug the removable 2-terminal block from the device power connector and then connect the removable 2-terminal block to the external power supply.

   **NOTICE**
   ***Check the supply voltage level and polarity before connecting the power supply:***
   – ***the device fails to operate properly with no risk of damage if the supply voltage is below 9 V.***
   – ***there is a high risk of the device damage if the supply voltage is above 30 V.***
   – ***the device does not turn on if the supply voltage polarity is reversed.***

3. Plug the removable 2-terminal block into the device mating power connector.

   **NOTICE**
   ***If the device is brought from a cold to a warm environment, condensation may form inside the device. To avoid damage to the device, keep the device in the warm environment for at least 1 hour before powering on.***

4. Apply the power supply voltage to the device.
5. Make sure that there are no errors indicated by the device (see *Section 7.1* and *Section 7.9.2*).
6. Start ***ALP*** and make sure that the device is detected correctly. Using ***ALP***, set the date and the time required (please refer to *Section 4.12*).
7. Remove the supply voltage from the device and then disconnect the USB cable.
8. Connect the USB cable to the device and apply the supply voltage again. Check the date and the time set. Replace the RTC battery if the date and time set are not retained being caused by RTC reset. Please refer to *Section 8.2* for the RTC battery replacement.
9. Using ***ALP***, configure the device and create a user program. The completed project can be transferred to the device memory using the menu item ***Device > Transfer application to device***. The user program is retained in the nonvolatile memory of the device and starts after powering the device on or its reset. See ***ALP*** Help for detailed information about configuration.
10. Remove the supply voltage from the device.
11. Connect cables from external devices and sensors to the removable terminal blocks.
12. Plug the removable terminal blocks into the appropriate mating connectors of the device. The terminal blocks layout and the assignment of the terminals are given in *Appendix A.*

If the user program does not start after the device is powered on, it is necessary to repeat transfer of the user program to the device. If the user program was transferred to the device with errors, it may cause the device incorrect operation. In this case, it is necessary to stop the user program by setting the RUN/STOP switch to STOP position.

## 4.2 Inputs

### 4.2.1 Analog inputs

Open the node ***Device > Analog inputs*** in the configuration dialog to access the analog inputs configuration menu (see the figure below).

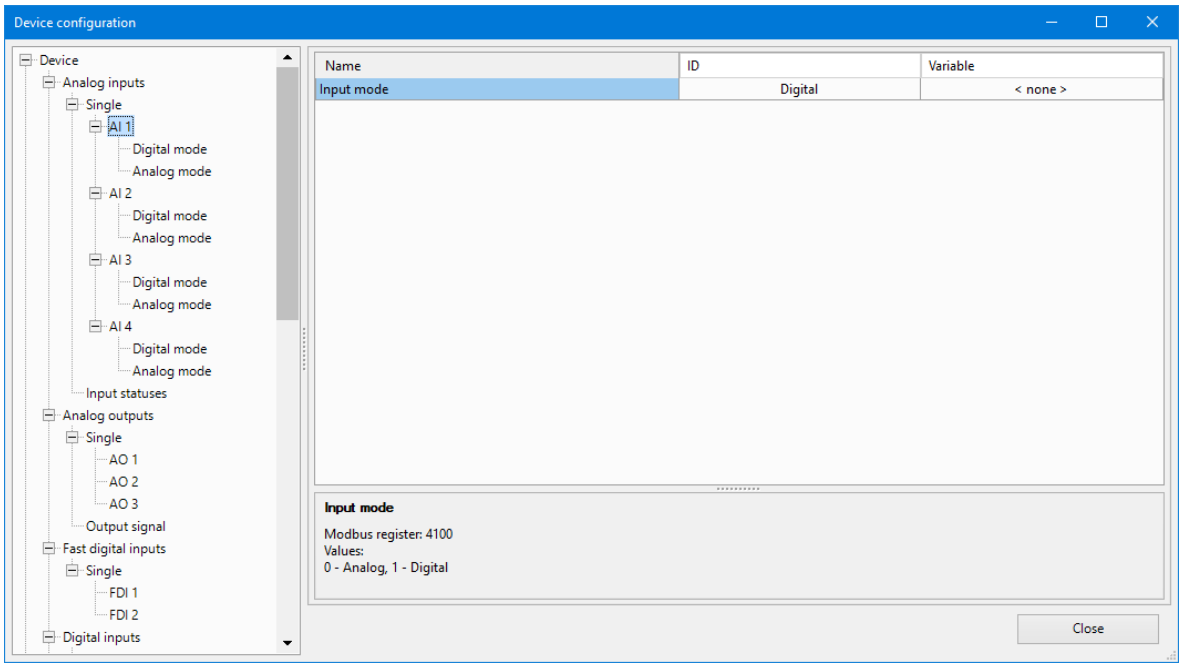



*Fig. 4.1 Analog input, mode selection in ALP*

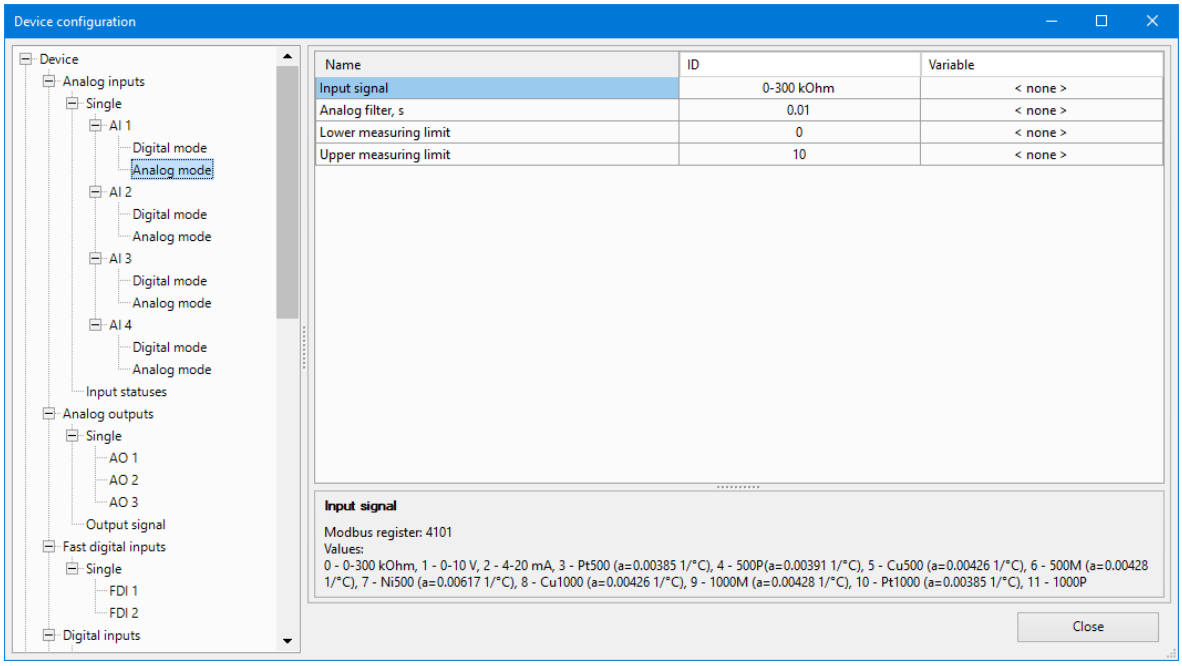



*Fig. 4.2 AI analog mode parameters in ALP*

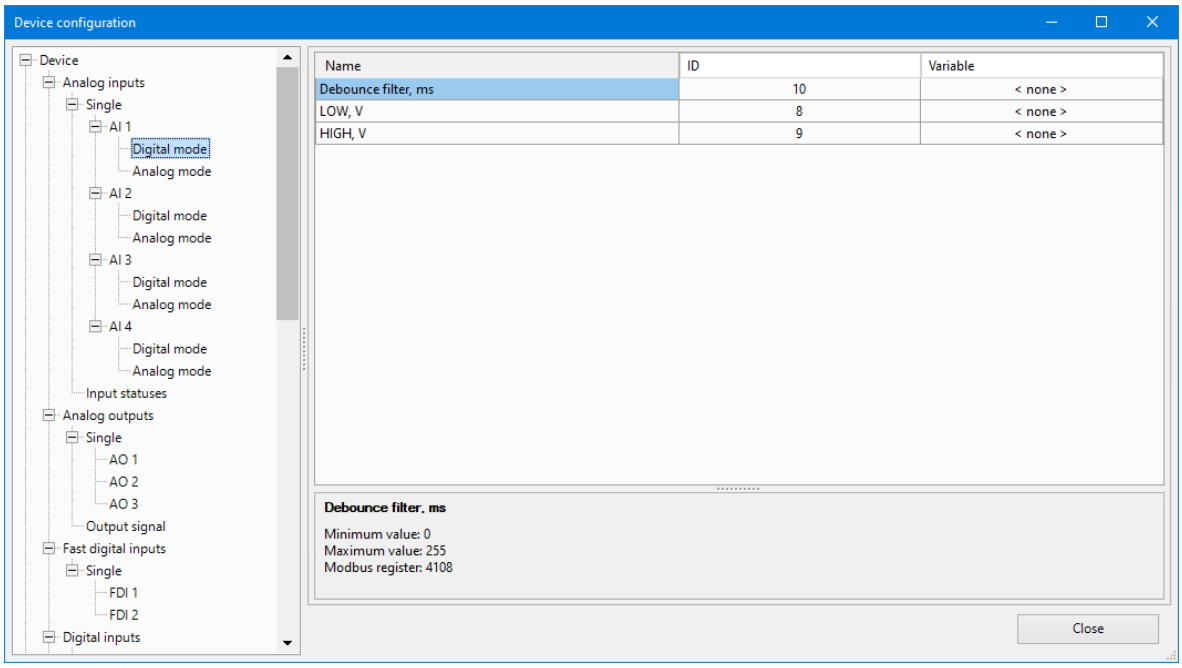



*Fig. 4.3 AI digital mode parameters in ALP*

*Table 4.1 AI configuration parameters*

| Parameter | Description |
|---|---|
| ***Input mode*** | AI mode selection: analog mode or digital mode |
| **Analog mode configuration parameters** | |
| ***Input signal*** | Selection of the input signal type from sensors:<br>0...300 kΩ<br>0...10 V<br>4...20 mA<br>RTD and NTC/PTC |
| ***Analog filter*** | The time constant of the built-in digital anti-aliasing filter. The time constant value sets the time of the input signal processing. The more the time constant value, the better the input channel noise immunity. On the other hand, increasing the time constant value reduces the input channel bandwidth, thus resulting in a slower device response while processing fast changing analog input signals. |
| ***Lower measuring limit*** | Minimum signal strength of the input signal measured. It is used for scaling the input signal. |
| ***Upper measuring limit*** | Maximum signal strength of the input signal measured. It is used for scaling the input signal. |
| **Digital mode configuration parameters** | |
| ***Debounce filter*** | The debouncing time of the built-in digital debouncing filter. The value sets a bandwidth limit for the digital input signal processing. The more the value, the better the input channel noise immunity. On the other hand, increasing the value reduces the input channel bandwidth, thus resulting in a slower device response while processing fast changing digital input signals. |
| ***LOW*** | Switching threshold from high level to low level |
| ***HIGH*** | Switching threshold from low level to high level |

#### 4.2.1.1 Analog mode

In order to ensure measuring the 4...20 mA current signals, each analog input channel of the device is equipped with the built-in 121 Ω shunt resistor.

The device supports the scaling of the measured 0...10 V and 4...20 mA input signals by setting the lower and upper measuring limit parameters. Thus, as the scaling is applied, the measured input signals are displayed in initial units of input parameters measured by sensors, e. g. atm (kg/cm²), kPa, etc. Please refer to the example below for the explanation of the signal scaling.

***Example:***
**Example of the input signal scaling**
Given the following:
– the used sensor: 4...20 mA output pressure sensor having input measurement range of 0... 25 atm,
– the “Lower measuring limit” parameter value set: 0.00,
– the “Upper measuring limit” parameter value set: 25.00,
the analog input signal is now measured to correspond to the unit of atm (see the figure below).

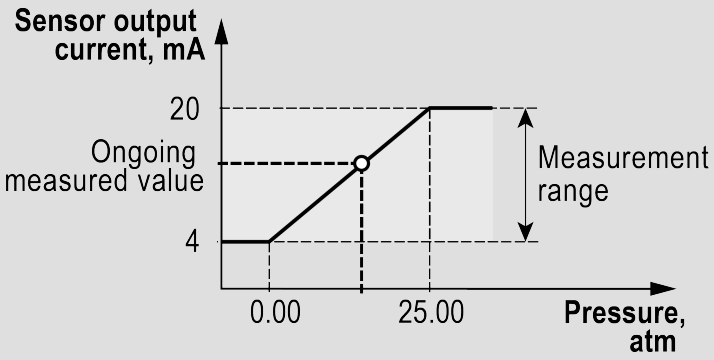



*Fig. 4.4 Example of the input signal scaling*

The device is operating with absolute parameter values (FLOAT32) of parameters.
The resistive signals are measured by the 2–wire resistance measurement method, for this reason the additional measurement error caused by the resistance of sensor wires must be taken into account. The value of the additional measurement error depends on the used sensor type and the length of the sensor wires. A correction of the additional measurement error must be implied in the user program.

### 4.2.1.2 Digital mode

The input operates as a comparator with parameters ***LOW*** and ***HIGH*** which determine the hysteresis (see the figure).

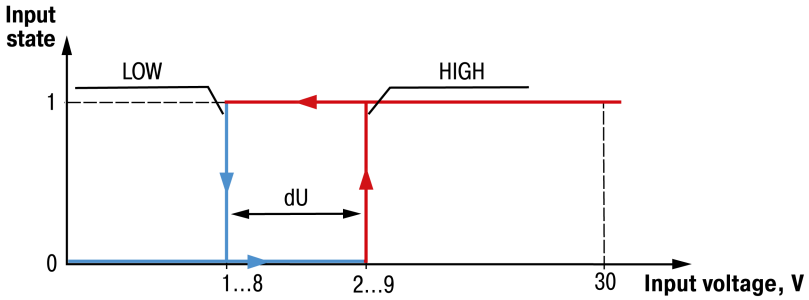



*Fig. 4.5 Analog input, digital mode diagram*

To avoid the ambiguity of determining the input state, the parameter ***HIGH*** must be set higher than the parameter ***LOW*** by at least 0.5 V.
The input state will not change if the input voltage is within the dU interval. The input state will change:
– from **logic low** to **logic high** only if the input voltage reaches the parameter ***HIGH*** set value.
– from **logic high** to **logic low** only if the input voltage reaches the parameter ***LOW*** set value.

### 4.2.2 Digital inputs

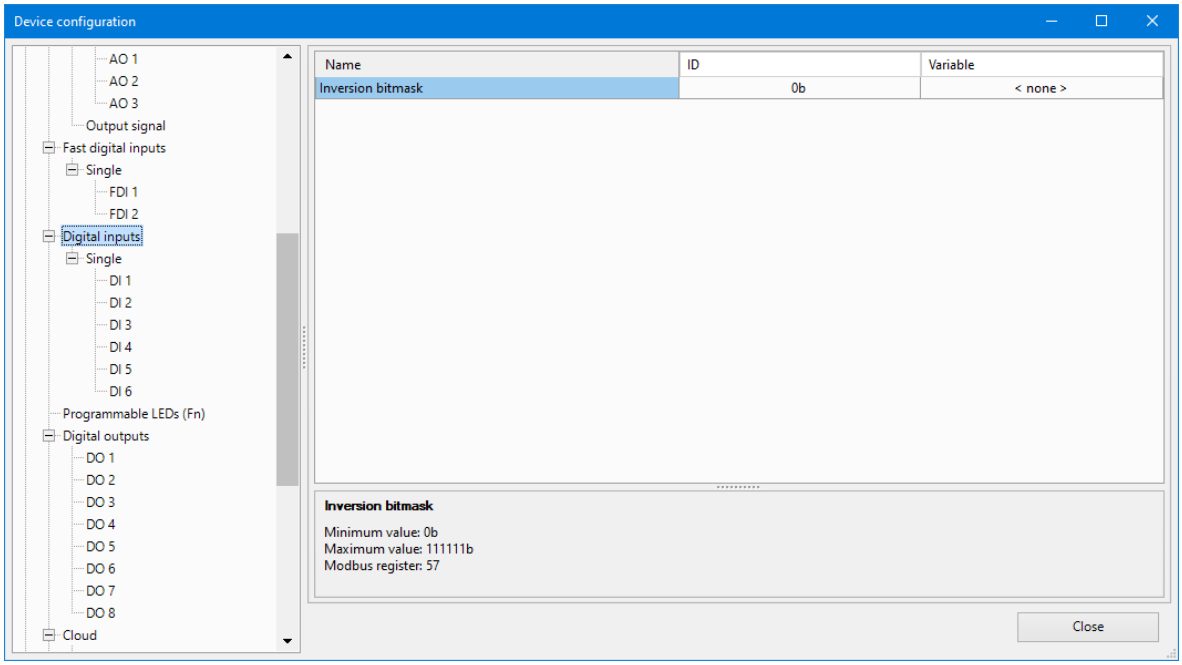



*Fig. 4.6 DI inversion configuration parameter*

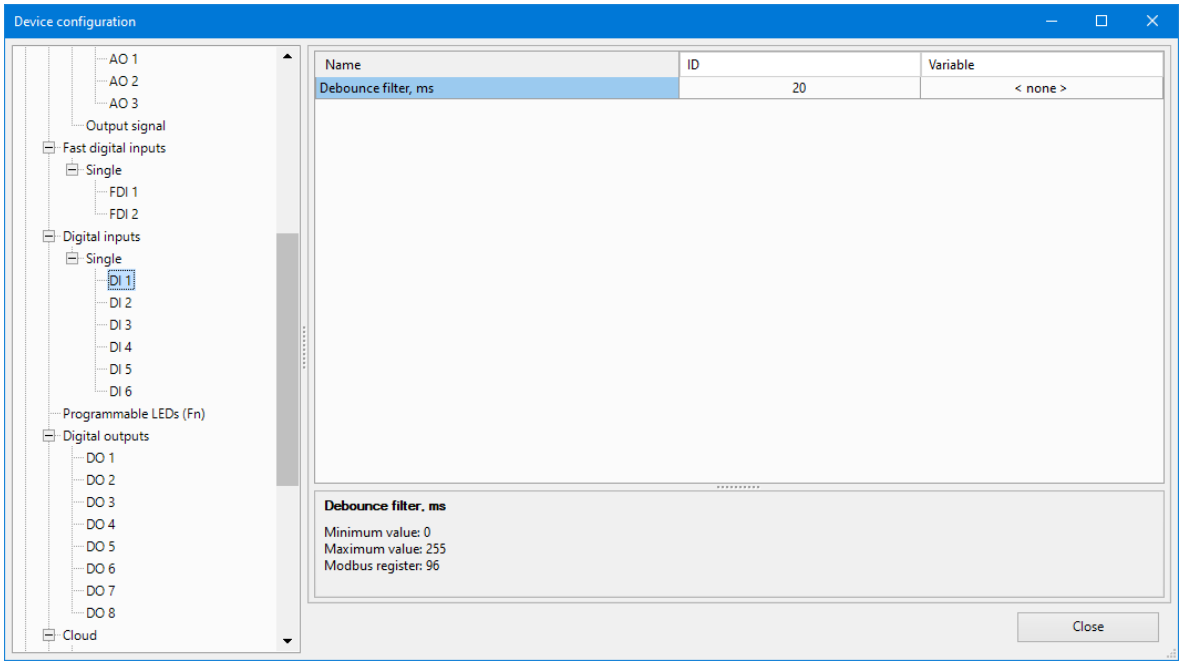



*Fig. 4.7 DI debounce filter configuration parameter*

*Table 4.2 DI configuration parameters*

| Parameter | Description |
|---|---|
| ***Inversion bitmask*** | Inversion bitmask is to invert single or multiple digital inputs by setting the bitmask variable value. |
| ***Debounce filter*** | The debouncing time of the built-in digital debouncing filter. The value sets a bandwidth limit for the digital input signal processing. The more the value, the better the input channel noise immunity. On the other hand, increasing the value reduces the input channel bandwidth resulting in a slower device response while processing fast changing digital input signals. |

### 4.2.3 Fast digital inputs

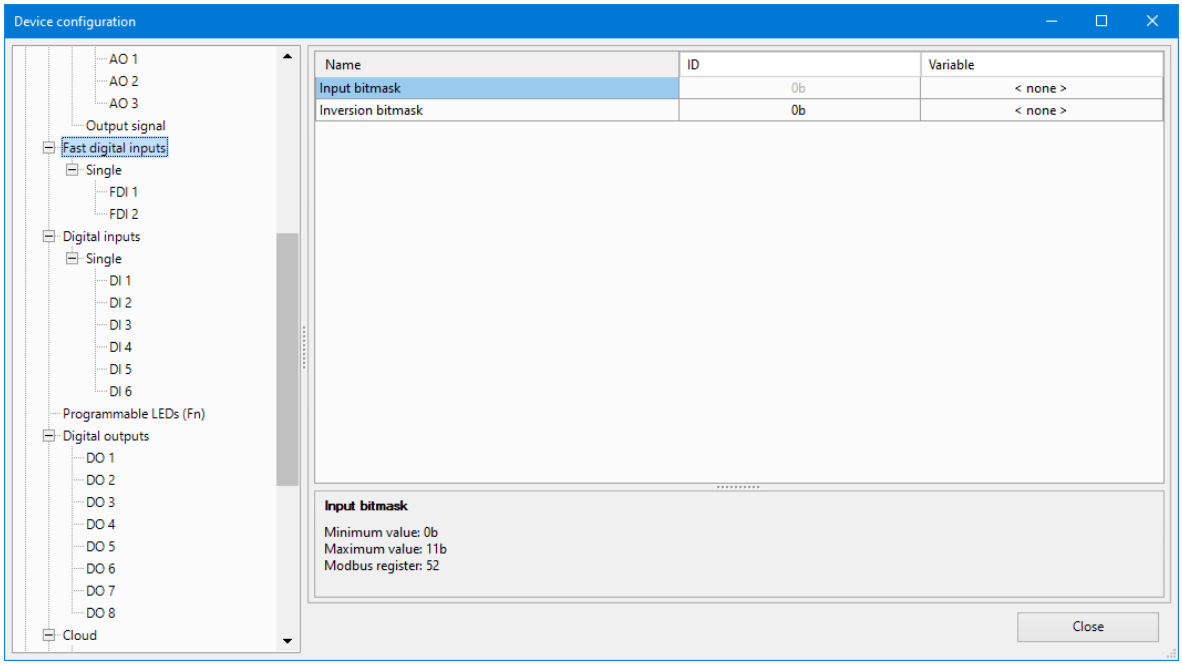



*Fig. 4.8 FDI configuration parameters*

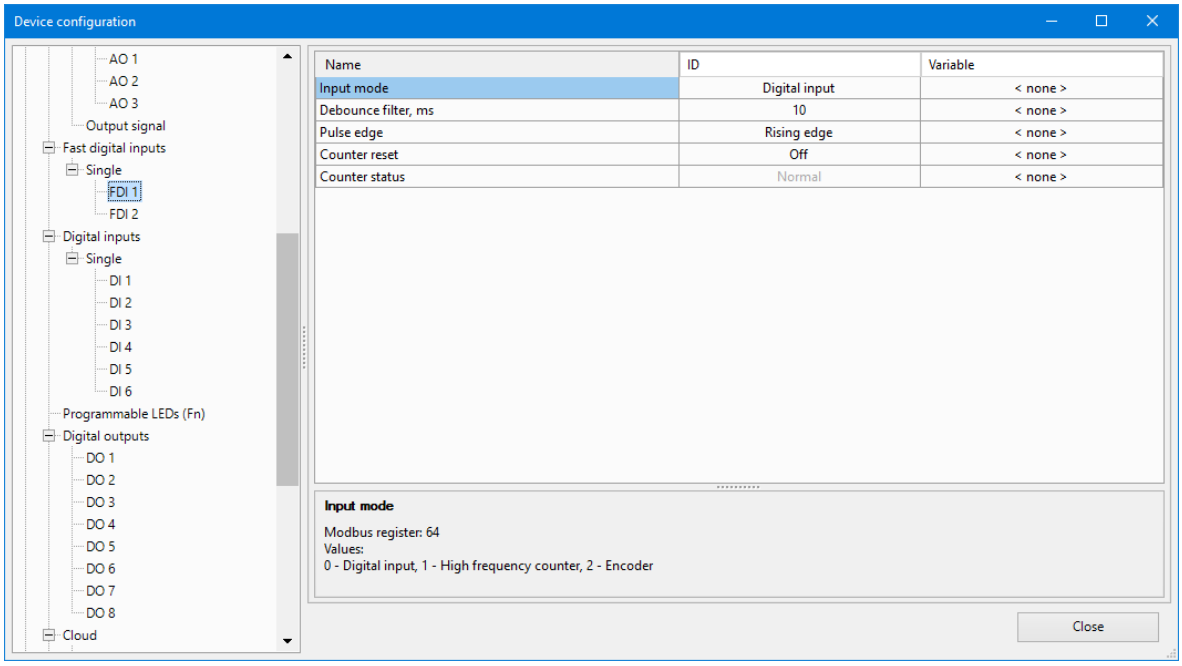



*Fig. 4.9 FDI modes configuration*

*Table 4.3 FDI modes configuration parameters*

| Parameter | Value | Description |
|---|---|---|
| ***Input mode*** | 0 – Digital input | The input is configured for the detection of the input logic level |
| | 1 – High frequency counter | The input is configured for the input pulses counting |
| | 2 – Encoder | The input is configured for the encoder signals processing |
| ***Debounce filter**** | from 0 to 255 ms | The parameter is applicable only when the input is configured as digital input. The |

| Parameter | Value | Description |
|---|---|---|
| | | parameter is to set debouncing time of the built-in digital debouncing filter. The default value is 10 ms. |
| ***Pulse edge*** | 0 – Rising edge | Pulses are triggered by their rising edge |
| | 1 – Falling edge | Pulses are triggered by their falling edge |
| ***Counter reset*** | On | Counter forced reset every 10 ms<br>**NOTE**<br>***The counter register is resetting itself if overflowed.*** |
| | Off | No counter forced reset |
| **NOTE**<br>**** It is not recommended to use debounce filter for input signals with a frequency above 40 Hz and a duty cycle of 50 % or less. A useful signal can be missed.*** | | |

**NOTE**
***The input counters retain their values after the device restart.***

Two-channel encoders (without Z channel) with the maximum signal frequency of 100 kHz can be connected to the fast digital inputs.
The total number of pulses is stored in a 32-bit register, taking into account the direction of rotation after a zero crossing. When the direction of rotation changes (e.g. from positive to negative), pulses are counted with the opposite sign (in this case, subtracted).

## 4.3 Outputs

### 4.3.1 Digital outputs

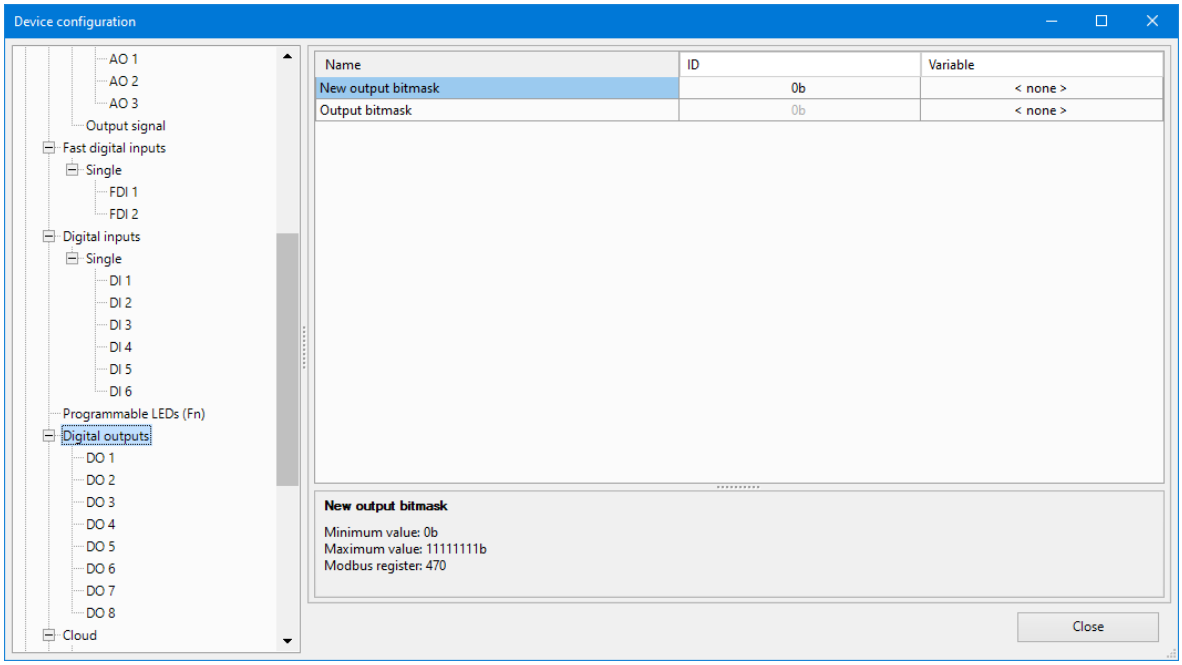



*Fig. 4.10 DO configuration parameters*

The safe state can be assigned for each of the device digital outputs as well as for each of the PRM extension module digital outputs (if used).

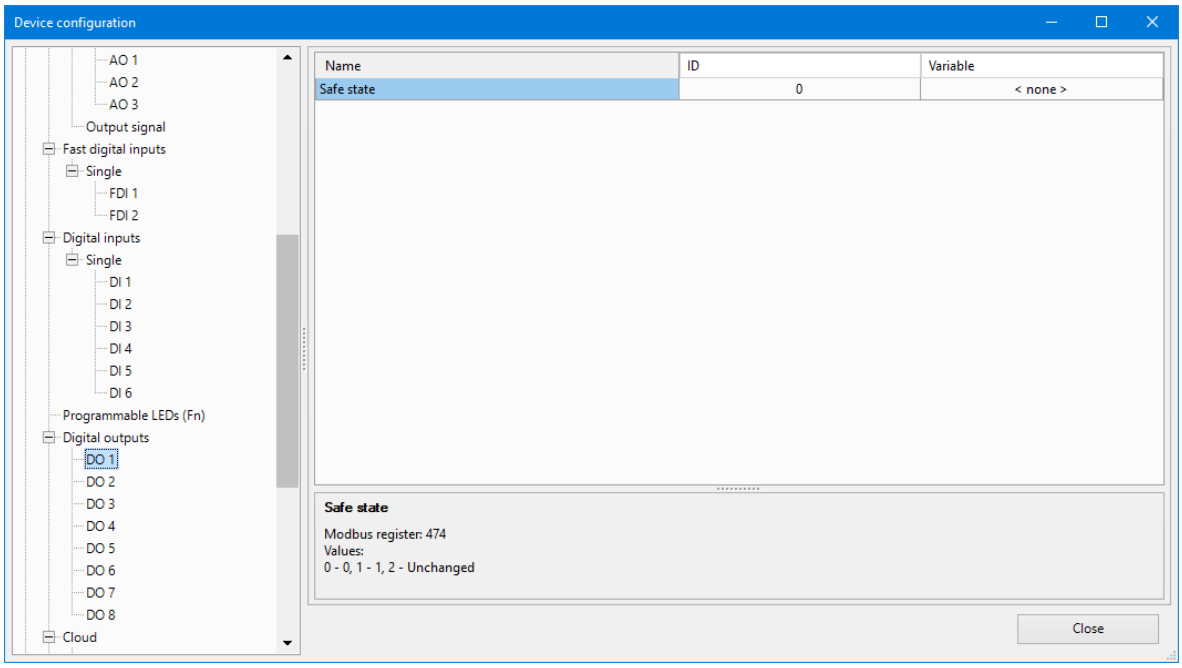



*Fig. 4.11 DO safe state configuration parameters*

*Table 4.4 DO safe state configuration parameters*

| Parameter | Description |
|---|---|
| ***Safe state*** | The parameter is to assign an output state for the digital outputs when the device does not receive network Master commands within the time-out period (communication loss). The following values are to be set to assign appropriate output states:<br>value “0” – 0 (digital output is set to 0).<br>value “1” – 1 (digital output is set to 1).<br>value “2” – unchanged (digital output stays unchanged). |

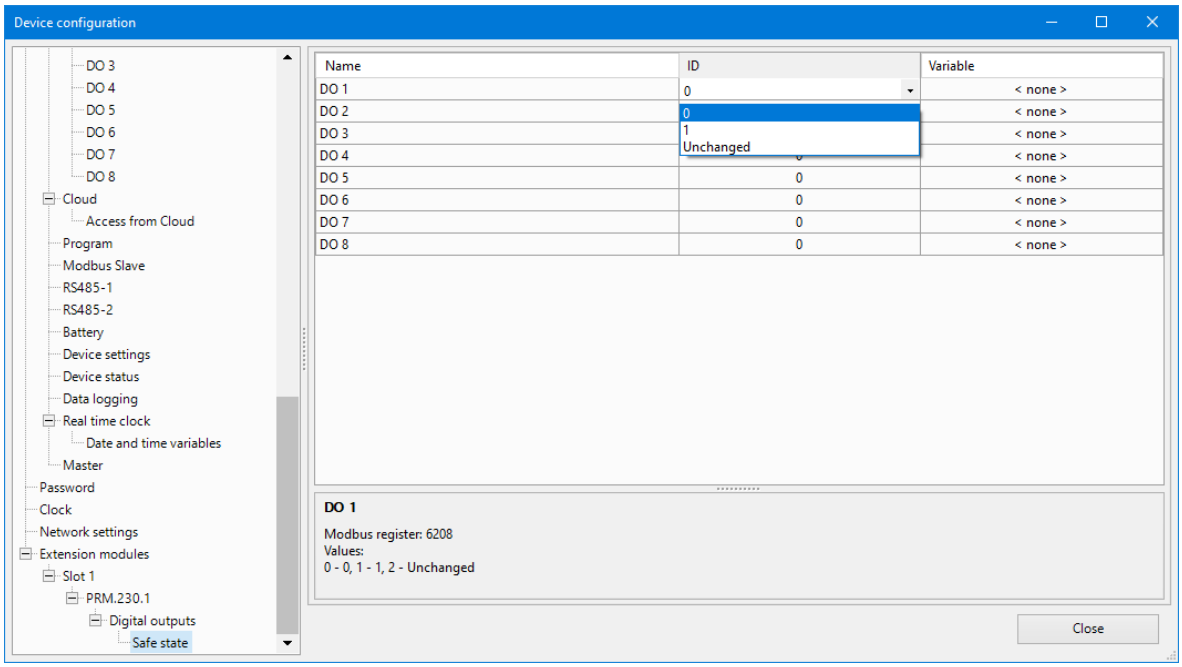



*Fig. 4.12 PRM extension module DO safe state configuration parameters*

*Table 4.5 PRM extension module DO safe state configuration parameters*

| Parameter | Description |
|---|---|
| ***Safe state*** | The parameter is to assign an output state for the PRM extension module digital outputs when communication lost. The following values are to be set to assign appropriate output states:<br>value “0” – 0 (PRM digital output is set to 0).<br>value “1” – 1 (PRM digital output is set to 1).<br>value “2” – unchanged (PRM digital output stays unchanged). |

### 4.3.2 Analog outputs

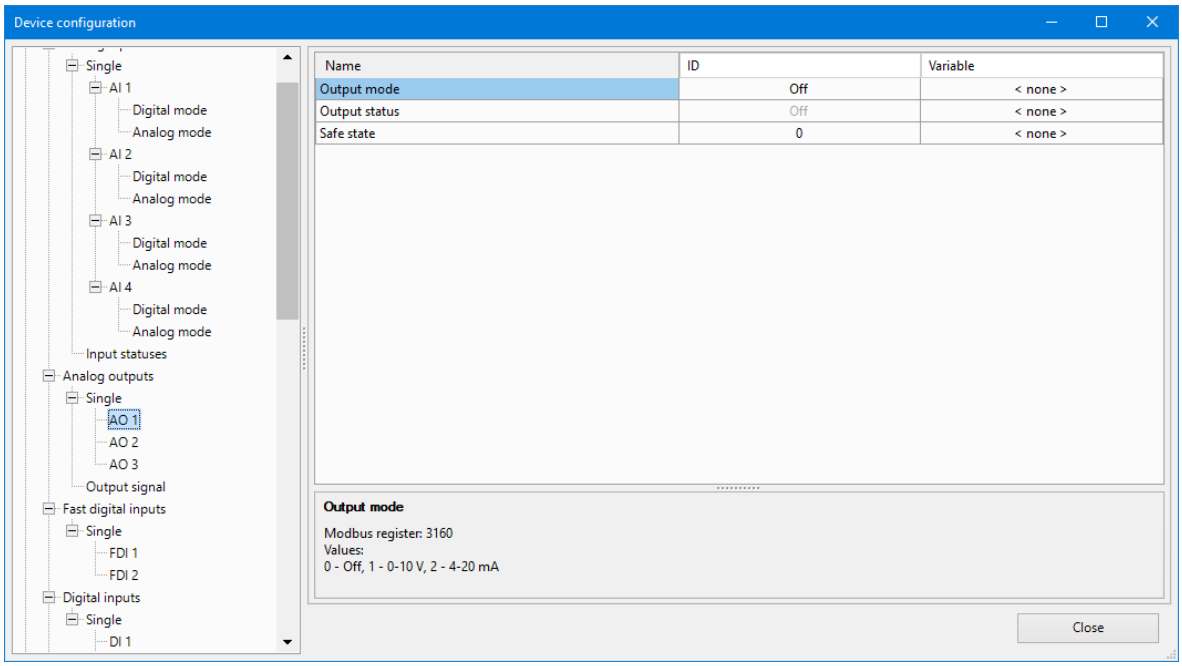



*Fig. 4.13 AO parameters in ALP*

*Table 4.6 AO configuration parameters*

| Parameter | Description |
|---|---|
| ***Output mode*** | The parameter is to select the type of the analog output signal. The following values are to be set to select appropriate signal types:<br>value “0” – Off (no analog output signal).<br>value “1” – 0...10 V (the analog output is configured for 0...10 V output).<br>value “2” – 4...20 mA (the analog output is configured for 4...20 mA output). |
| ***Output status*** | The read-only diagnostic parameter. The following values are available for reading:<br>0 – Off<br>1 – Normal operation<br>2 – No connection<br>3 – Failure |
| ***Safe state*** | The safe state parameter is to set a required analog output signal level when the device does not receive network Master commands within the time-out period (communication loss). The safe state parameter value is set within the range from 0 to 1. |

To control an analog output, a floating-point value of type FLOAT32 within the range from 0.0 to 1.0 has to be assigned to it in the user program.

***Example:***

Given the analog output is configured for 4...20 mA output and the value of 0.5 is applied, the output current is 12 mA.

***Example:***
Given the analog output is configured for 0...10 V output and the value of 0.5 is applied, the output voltage is 5 V.

### 4.4 LED indicators

PR205 has two programmable LED indicators: F1 and F2. A variable can be assigned to the states of the LED indicators to control the LED states in the user program.

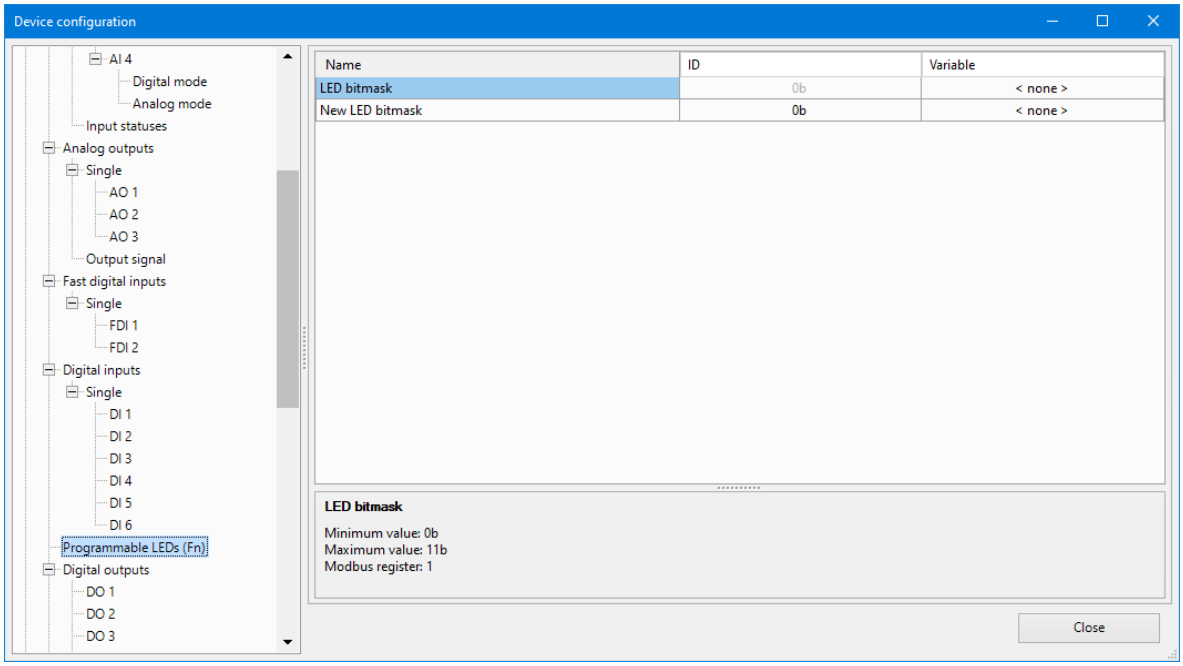



*Fig. 4.14 LED parameters in ALP*

*Table 4.7 User defined LEDs*

| Parameter | Description |
|---|---|
| LED bitmask | Bitmask, little endian byte order |

### 4.5 Data logging

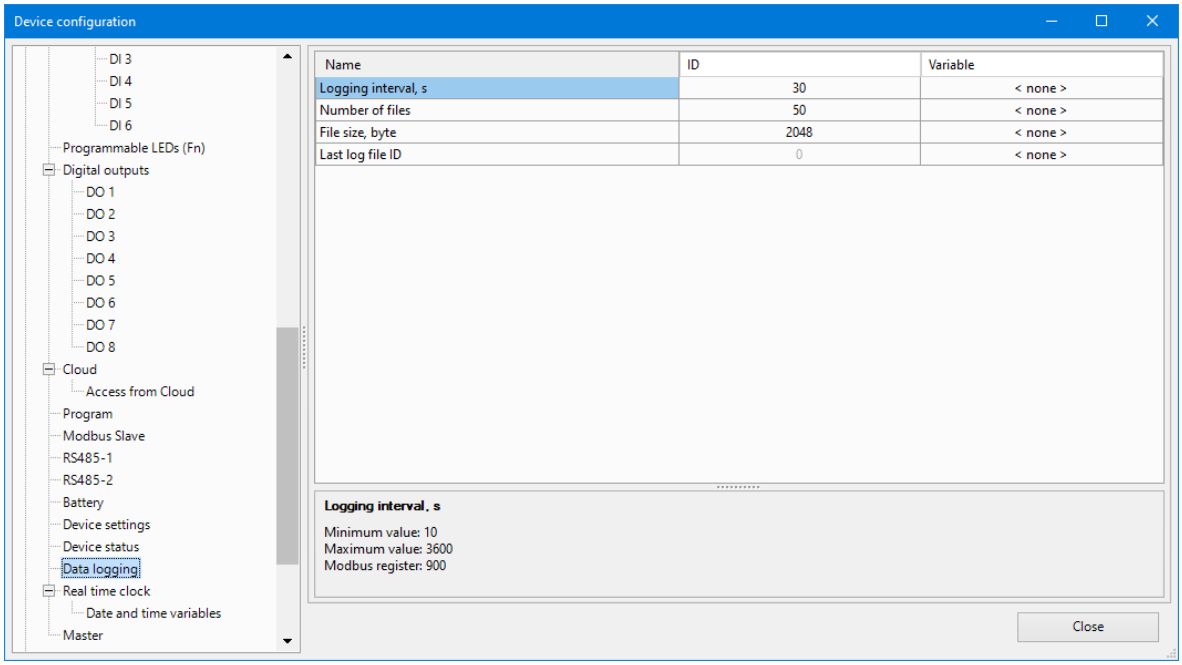



*Fig. 4.15 Data logging parameters in ALP*

*Table 4.8 Data logging parameters*

| Parameter | Description |
|---|---|
| ***Logging interval*** | Time interval in seconds at which the values of the variables selected for logging are recorded |
| ***Number of files*** | Maximum number of files the archive consists of |
| ***File size, byte*** | Log file size in Bytes |
| ***Last log file ID*** | ID of the last written file |

### 4.6 Working with akYtec ToolPro

Configuration with ***akYtec ToolPro*** is performed "in real time". The configuration parameters can be read from the device, viewed, modified and saved to it. For configuration over Ethernet, the device must be powered on. When being configured over USB, the device is powered by USB and the main power supply is not required.

1. Start ***akYtec ToolPro***.
2. Click menu item ***Add device***
3. Select interface in a drop-down list ***Interface***:
   – Ethernet – for connection via Ethernet
   – STMicroelectronics Virtual COM Port – for connection via USB or RS485

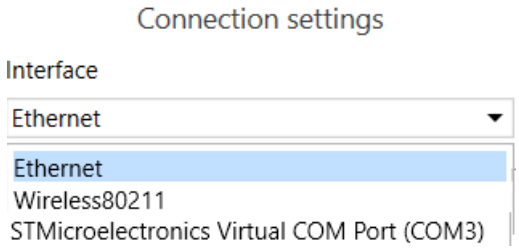



*Fig. 4.16 Interface selection*

Further steps for finding device in a network depend on the selected interface.

1. To find a device in an Ethernet network:
   – Select radio button ***Find device***
   – Specify the device IP address (default – 192.168.1.99)
   – Click ***Find***. The found device will be displayed in the right field
   – Select the device and press ***OK***
   – If the device is password protected (*Sect. 4.7.1*), enter the correct password
2. To find a device connected over USB or RS485:
   – Select ***akYtec Auto Detection Protocol*** in the drop-down list ***Protocol***.

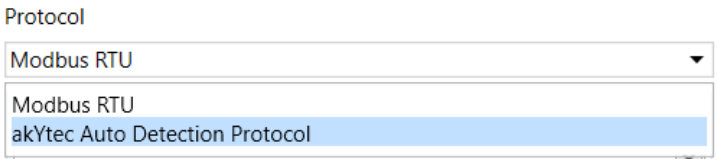



*Fig. 4.17 Protocol selection*

   – For USB, select ***Connection setup*** – ***Auto***
   – For RS485, select ***Connection setup*** – ***Manually*** and specify the connection parameters:
      – Baud rate – **9600** kbit/s
      – Data bits – **8**
      – Parity – **none**
      – Stop bits – **1**.
   – Select radio button ***Find device***.
   – Specify the device address (default — **16**).
   – Click the button ***Find***. The found device will be displayed in the right field.
   – If the device is password protected (*Sect. 4.7.1*), enter correct password

## 4.7 Password and archives

### 4.7.1 Password

You can protect your configuration parameters and user program with a password. There is no password by default.

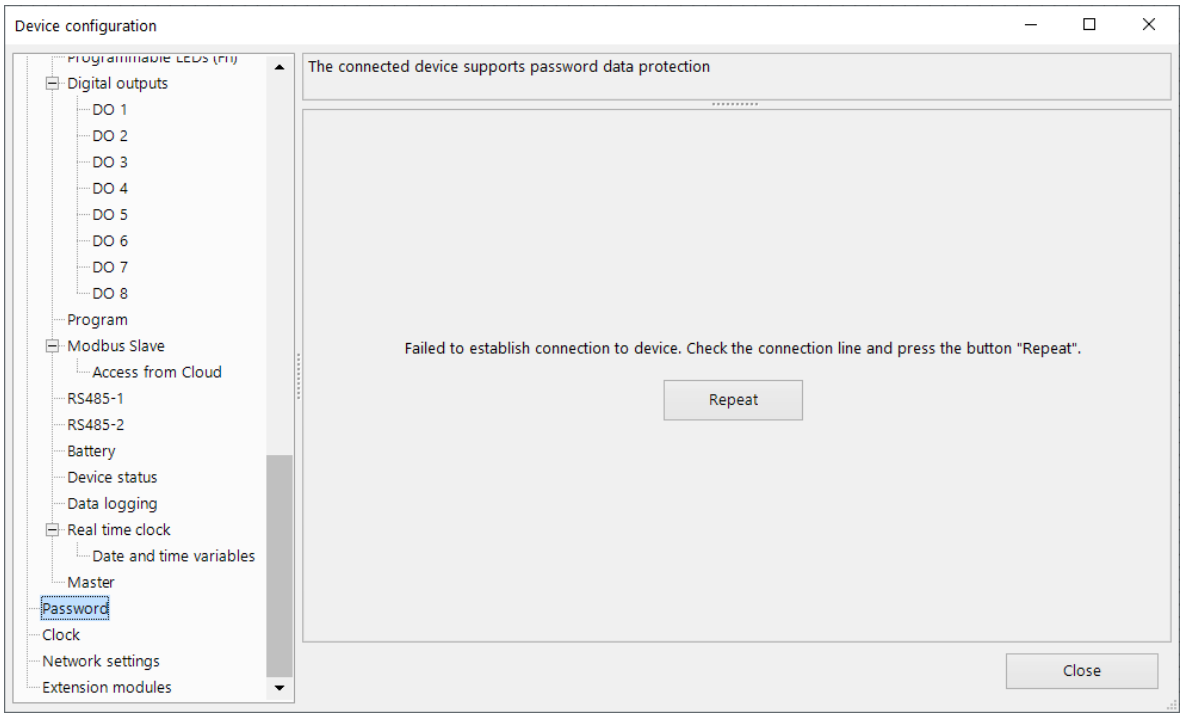



*Fig. 4.18 Password setting in ALP*

If you forgot the password, restore the factory setting (*section 4.13*).

### 4.7.2 Archives

The device is equipped with a built-in flash memory, formatted for the cryptographic file system which supports the file encryption.
*Data Encryption Standard* algorithm is used for data encryption. *Superkey* type key is used as a key. The initialization vector is generated by a hash function. Please see details in *Appendix C.*
By default, the following data is stored in the log file:

– Battery status.
– Device status - service information for the technical support.

An archive is saved as a set of encrypted log files. The log file consists of a set of records separated by Newline (0x0D0A). Each record corresponds to one parameter and consists of fields separated by semicolon. The record format is represented in the table below.

*Table 4.9 Record format*

| Field | Type | Size | Description |
|---|---|---|---|
| Timestamp | binary | 4 Bytes | In seconds, beginning from 00:00 01.01.2000 (UTC+0) |
| Separator | string | 1 Byte | Semicolon (;) |
| UID (parameter ID) | string | 8 Bytes | String of HEX characters with leading zeros |
| Separator | string | 1 Byte | Semicolon (;) |
| Parameter value | string | parameter depending | String of HEX characters with leading zeros |
| Separator | string | 1 Byte | Semicolon (;) |
| Parameter status | binary | 1 Byte | 1 – value correct<br>0 – value incorrect, further processing not recommended |
| Newline | binary | 2 Bytes | \n\r (0x0A0D) |

Data logging is a cyclic process.

The data logging parameters (the logging interval, the number of files and the maximum file size) can be set by user in ***ALP*** (see *section 4.5*). When the archive overflows, the data will be overwritten, starting with the oldest record in the oldest file.
The archive files are time-tagged by the device built-in RTC.
The time zone is not contained in the file but can be read from the parameter ***Time zone***.
The archive can be read with ***ALP***.
It can be also read via Modbus using the read function 20 (0x14) Read File Record. For details see *Modbus specifications*.

**CAUTION**
***The last archive file may be not retained on powering the device down.***

## 4.8 Network

### 4.8.1 General

Network parameters can be set with ***akYtec Tool Pro*** or in ***ALP***.
The device operates:

– under the Modbus TCP protocol (Slave / Master) over the Ethernet interface (see *Section 4.8.4*).

**NOTE**
***The device supports four threads per request when operating under Modbus TCP.***

– under the Modbus RTU protocol (Slave / Master) over the RS485 interface (see *Section 4.8.3* and *Section 6.9*).

### 4.8.2 Network parameters

#### 4.8.2.1 Ethernet – Modbus TCP

**NOTICE**
***In order to apply new Ethernet settings, restart the device. Disconnect the device from the USB interface before the device is restarted.***

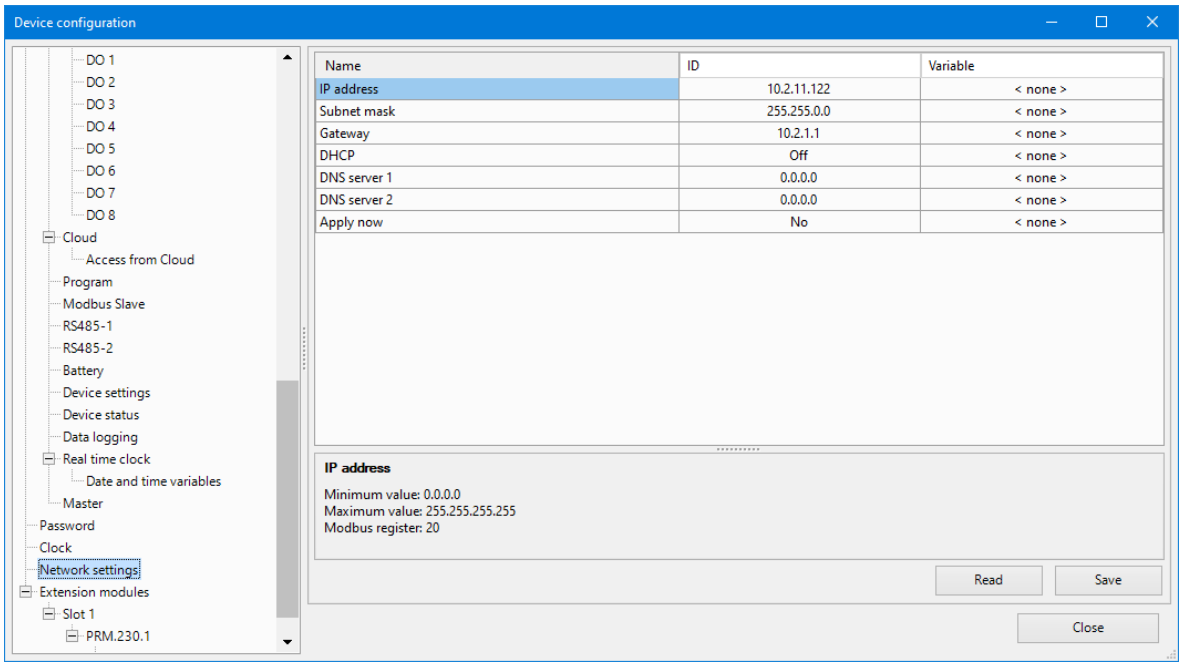



*Fig. 4.19 Ethernet parameters in ALP*

There are the two types of the IP address: a static IP address and a dynamic IP address. The static IP address can be also set by using the service button (in this case set the ***DHCP*** parameter value to ***Service button***).

The network parameters listed in the table below must be set to ensure communication over the Ethernet.

*Table 4.10 Ethernet parameters*

| Parameter | Description |
|---|---|
| ***IP address*** | IP address: it can be static IP or dynamic IP. The IP address default factory setting is **192.168.1.99** |
| ***Subnet mask*** | IP address recognition area in the subnet. The default factory setting is **255.255.0.0** |
| ***Gateway*** | IP address of the gateway. The default factory setting is **192.168.1.80** |
| ***DNS server 1*** | Used to translate the host name to the numerical IP addresses. Set the parameter to the value: **8.8.8.8** |
| ***DNS server 2*** | |
| ***New IP address*** | Enter new value if required |
| ***New subnet mask*** | |
| ***New gateway*** | |
| ***DHCP*** | DHCP mode settings. The following values are available for setting:<br>0 – On (DHCP mode is on).<br>1 – Off (DHCP mode is off).<br>2 – Service button (DHCP mode is one-time set by the service button). |

To configure the Ethernet interface, proceed as follows:

1. open node ***Network settings*** in ***ALP*** configuration dialog.
2. set parameters ***New IP address***, ***New subnet mask***, ***New gateway***.

The DHCP parameter must be set to “Off”.

#### 4.8.2.2 RS485

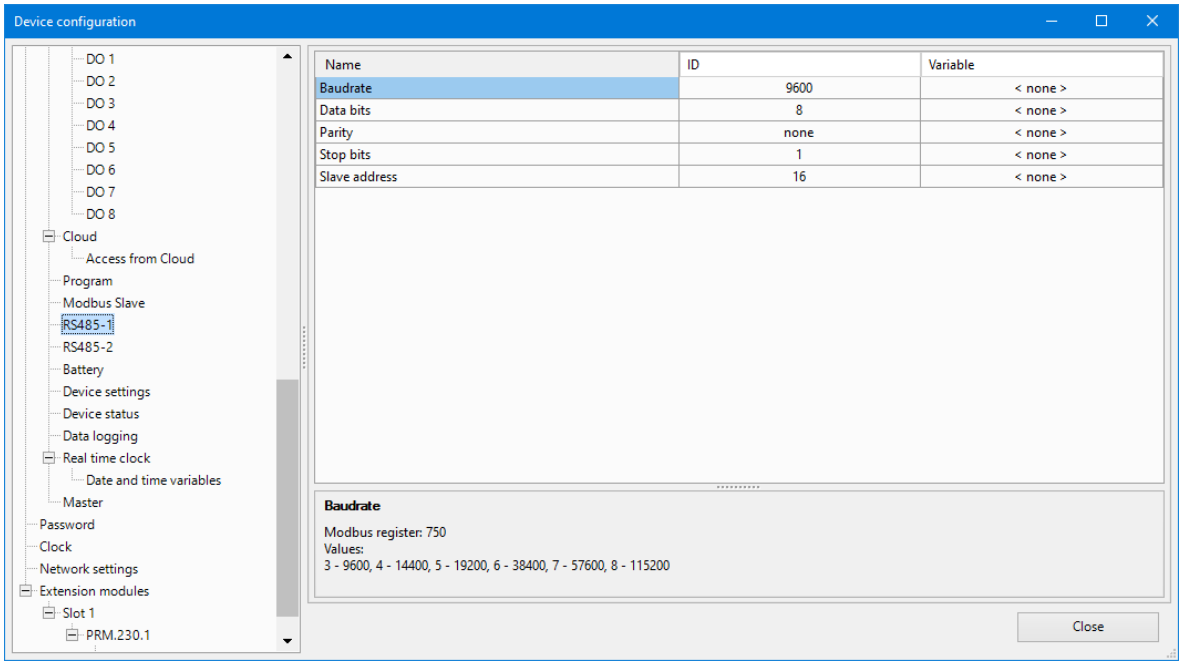



*Fig. 4.20 RS485 configuration parameters*

*Table 4.11 RS485 parameters*

| Parameter | Description |
|---|---|
| ***Baud rate*** | The parameter is to set the COM port baud rate. The following values can be set:<br>3 – 9600 bit/s<br>4 – 14400 bit/s |

| | |
|---|---|
| | 5 – 19200 bit/s<br>6 – 38400 bit/s<br>7 – 57600 bit/s<br>8 – 115200 bit/s |
| ***Data bits*** | Size of the data packet. The following values can be set:<br>0 – 8 bit<br>1 – 7 bit |
| ***Stop bits*** | Number of stop bits. The following values can be set:<br>0 – no<br>1 – odd<br>2 – even |
| ***Parity*** | Parity settings. The following values can be set:<br>0 – 1 stop bit<br>1 – 2 stop bits |
| ***Slave address*** | The device RS485 network address. The address set by default: 16 |

### 4.8.3 Modbus RTU and Modbus ASCII

Depending on the used RS485 interface protocol, the device can operate either in the Modbus RTU (Master/Slave) mode or the Modbus ASCII (Master/Slave) mode. The device recognizes Modbus RTU and Modbus ASCII protocols automatically. Please, refer to *Section 4.8.5* for configuration parameters.

### 4.8.4 Modbus TCP

Modbus TCP protocol is used only for communication over the device Ethernet port. Please, refer to *Section 4.8.2.1* for the Modbus TCP configuration parameters.

**NOTE**
***Disable DCHP mode when:***
– ***there is no DCHP server in the local network.***
– ***the point-to-point network is used to connect the device and PC.***

### 4.8.5 Modbus Master and Slave modes

**Master mode**

Only one Master device is allowed in the network. The following functions are supported when the device operates in the Master mode:
– reading by timer.
– reading/writing by event.
– writing by change (set by default).
The device is capable to control up to 32 slave-devices including:
– up to 64 variables at one operation (either read or write operation).
– up to 32 variables at two simultaneous operations (both read and write operations).
Using the same addresses and variables' names for each of the controlled devices is allowed.
Network variables' memory size for the device Master mode is 128 Bytes.
In order to ensure the device communication in the Master mode, appropriate controlled devices must be added and configured in the Modbus Master configuration parameters as shown in the figure below.

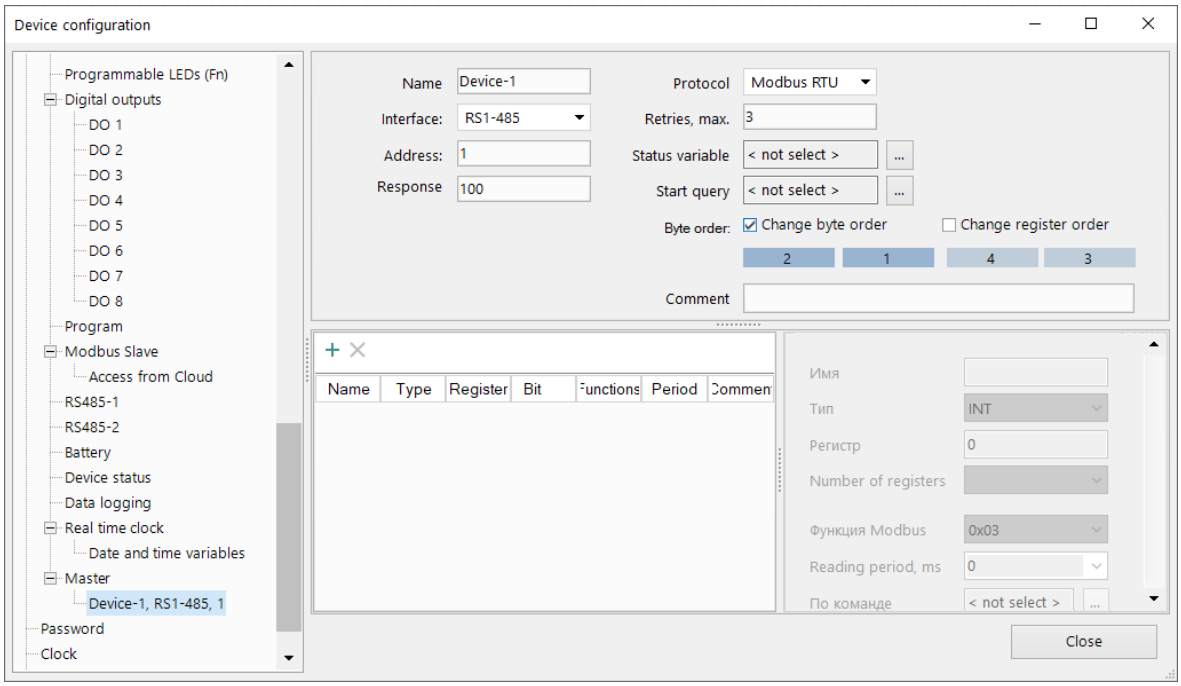



*Fig. 4.21 Modbus Master configuration parameters*

*Table 4.12 Modbus Master configuration parameters*

| Parameter | | Description |
|---|---|---|
| ***Name*** | | Setting a name of the controlled device to be displayed in the configuration tree |
| ***Interface*** | | Setting the communication interface (RS485 or Ethernet) |
| ***Address*** | | The controlled device network address |
| ***Response*** | | Setting the request timeout. The request is failed if there is no response after the set timeout |
| ***Protocol*** | | Setting the communication protocol |
| ***Retries, max*** | | Maximum number of request retries followed by changing the controlled device status |
| ***Byte order*** | ***Change register order*** | Setting the sequential order of sending registers for the two-register variables. The least register first as the box unchecked |
| | ***Change byte order*** | Setting the sequential order of sending registers. The most significant byte first as the box checked |

For detailed description please refer to ALP user manual available on the homepage *www.akYtec.de*.

### Slave mode

The following functions are supported when the device operates in the Slave mode:

– reading data from the multiple flag, holding and input registers.
– reading data from the single flag, holding and input registers.
– writing data to the multiple flag, holding and input registers.
– writing data to the single flag, holding and input registers.

Use ALP to configure the device for the Slave mode operation. Please refer to the Modbus register map to ensure the device Slave mode communication. The Modbus register map is given in *Appendix B.*

Network variables' memory size for the device Slave mode is 2040 Bytes.

### 4.8.6 Safe mode timeout

The device supports entering the safe state as communication with the network Master is lost.

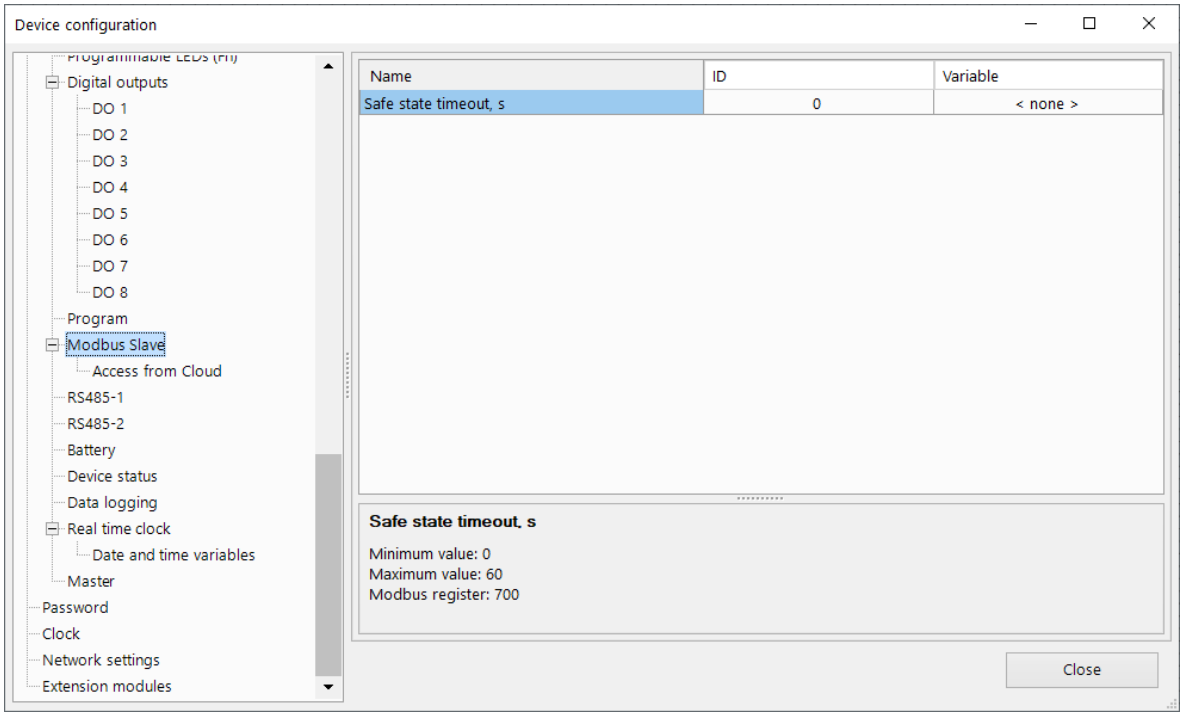



*Fig. 4.22 Safe mode timeout configuration parameters*

*Table 4.13 Safe mode timeout configuration parameters*

| Parameter | Description |
|---|---|
| ***Safe state timeout, s*** | The device enters the safe state as no commands received from the network Master after the set timeout. The device does not enter safe state upon communication loss if the parameter value is set to 0. |

### 4.8.7 Modbus application

*Table 4.14 Supported Modbus functions*

| Function | Code | Description |
|---|---|---|
| MODBUS_READ_HOLDING_ REGISTERS | 03 (0x03) | Reading data from one or more holding registers |
| MODBUS_READ_INPUT_ REGISTERS | 04 (0x04) | Reading data from one or more input registers |
| MODBUS_WRITE_SINGLE_ REGISTER | 06 (0x06) | Writing data to a single holding register |
| MODBUS_WRITE_MULTIPLE_ REGISTERS | 16 (0x10) | Writing data to multiple holding registers |
| MODBUS_READ_FILE_ RECORD | 20 (0x14) | Read record file as a set of records |
| MODBUS_WRITE_FILE_ RECORD | 21 (0x15) | Write record file as a set of records |

The bitmask parameters can be read by functions 0x03 and 0x01.

If you are using function 0x01 for reading, multiply the register number by 16 and add the number of the required bit to determine the start bit.

*Table 4.15 Basic registers*

| Parameter | Register | Size (Byte) | Type | Comments |
|---|---|---|---|---|
| Device name to appear for user (DEV) | 0xF000 | 32 | String | Win-1251 |
| Firmware version to appear for user (VER) | 0xF010 | 32 | String | Win-1251 |
| Series | 0xF020 | 32 | String | Win-1251 |
| Subseries | 0xF030 | 32 | String | Win-1251 |
| Hardware version | 0xF040 | 16 | String | Win-1251 |
| Additional information | 0xF048 | 16 | String | Win-1251 |
| Time and date | 0xF080 | 4 | Unsigned 32 | UTC in seconds, starting from 0:00 01.01.2000 |
| Time zone | 0xF082 | 2 | Signed short | Offset from Greenwich time in minutes |
| S/N | 0xF084 | 32 | String | Win-1251, 17 symbols used |

*Table 4.16 Basic data formats*

| Format | Number of registers | Size (Byte) | Description |
|---|---|---|---|
| Unsigned 16 | 1 | 2 | Unsigned integer |
| Unsigned 32 | 2 | 4 | |
| Signed 16 | 1 | 2 | Signed integer |
| Datetime 32 | 2 | 4 | UTC date/time in seconds, starting from 0:00 01.01.2000 |

*Table 4.17 Special data formats*

| Format | Number of registers | Size (Byte) | Description |
|---|---|---|---|
| Enum 1...Enum 37 | 1 | 1 | Specifies a selected parameter position in the list of parameters (e.g. the sensor type used with the analog inputs) |
| Float 32 | 2 | 4 | The format representing a real number |
| Unsigned 8 | 1 | 1 | Unsigned integer |
| String 48 | 3 | 6 | String of 6 characters |
| String 64 | 4 | 6 | String of 8 characters |
| String 128 | 8 | 16 | String of 16 characters |

Please refer to the Modbus register map for the list of the Modbus registers. The Modbus register map is given in *Appendix B.*.
When using two-register (or more) variables, the sending sequential order is as follows:

– the byte order: the most significant byte first.
– the register order: the least significant register first.

### 4.8.8 Modbus error codes

*Table 4.18 Modbus error codes*

| Code | Description |
|---|---|
| 01 | Function code received in the query is not recognized or allowed by slave |
| 02 | Data address of some or all the required entities are not allowed or do not exist in slave |
| 03 | Value is not accepted by slave |
| 04 | Unrecoverable error occurred while slave was attempting to perform requested action. The device is in error mode. |

| Code | Description |
|---|---|
| 05 | Slave has accepted the request and is processing it, but it takes time. This response is returned to prevent a timeout error in the master. |
| 06 | Slave is engaged in processing a long-duration command. Master should retry later. |
| 08 | Specialized use in conjunction with function codes 20 and 21. Slave detected a parity error in memory. Master can retry the request, but service may be required on the slave device. |

The data packages are processed according to the steps as follows:

1. The data package validity check is performed. The data package is ignored if the check fails.
2. The data package address (Slave ID) check is performed. The data package is ignored if the check fails.
3. The Modbus function check is performed.

If the received request specifies a not supported Modbus function (see *table 4.14* for supported Modbus functions), the error MODBUS_ILLEGAL_ FUNCTION is issued.
The description of the data and files errors is given in the tables below.

*Table 4.19 Processing data errors*

| Modbus function used | Error name | Possible reason causing the error |
|---|---|---|
| MODBUS_READ_ HOLDING_ REGISTERS | MODBUS_ILLEGAL_DATA_ ADDRESS | The number of the requested registers exceeds the maximum possible number (125).<br>The requested parameter does not exist. |
| MODBUS_READ_INPUT_ REGISTERS | MODBUS_ILLEGAL_DATA_ ADDRESS | The number of the requested registers exceeds the maximum possible number (125).<br>The requested parameter does not exist. |
| MODBUS_WRITE_SINGLE_ REGISTER | MODBUS_ILLEGAL_DATA_ ADDRESS | The size of the parameter to be written exceeds 2 Bytes.<br>Write access is denied for the parameter.<br>The function does not support the type of the parameter to be written.<br>The requested parameter does not exist.<br>The data types supported:<br>– signed and unsigned integer (up to 2 Bytes).<br>– enumerated. |
| | MODBUS_ILLEGAL_DATA_ VALUE | The parameter falls outside its upper or lower value limits |
| MODBUS_WRITE_ MULTIPLE_REGISTERS | MODBUS_ILLEGAL_DATA_ ADDRESS | The parameter to be written does not exist.<br>Write access is denied for the parameter.<br>The number of the registers to be written exceeds the maximum permissible number (123). |

| Modbus function used | Error name | Possible reason causing the error |
|---|---|---|
| | MODBUS_ILLEGAL_DATA_ VALUE | The terminal null symbol (\0) is missing in the string parameter.<br>The size of data requested is less than the size of the first or the last requested parameters.<br>The parameter falls outside its upper or lower value limits. |

*Table 4.20 Archive files' errors*

| Modbus function used | Error name | Possible reason causing the error |
|---|---|---|
| MODBUS_READ_FILE_ RECORD | MODBUS_ILLEGAL_ FUNCTION | Invalid data size (0×07 ≤ data length ≤ 0×F5) |
| | MODBUS_ILLEGAL_DATA_ ADDRESS | Reference type does not meet the specification.<br>The file to be read failed to open or does not exist. |
| | MODBUS_ILLEGAL_DATA_ VALUE | Failed to move to the file offset required. |
| | MODBUS_SLAVE_DEVICE_ FAILURE | File deletion error occurred when requested for deletion.<br>The size of the data requested is too large (exceeding 250 Bytes).<br>Invalid record number (exceeding 0×270F).<br>Invalid record length (exceeding 0×07A). |
| MODBUS_WRITE_FILE_ RECORD | MODBUS_ILLEGAL_ FUNCTION | Invalid data size (0×09 ≤ data length ≤ 0×FB). |
| | MODBUS_ILLEGAL_DATA_ ADDRESS | Reference type does not meet the specification.<br>The file to be written failed to open. |
| | MODBUS_SLAVE_DEVICE_ FAILURE | The requested file does not exist.<br>The requested file is read-only.<br>Failed to write the required number of bytes. |

### 4.9 Device status

The parameters of the device status and the device battery status are available in the ALP parameter tree dialogue.

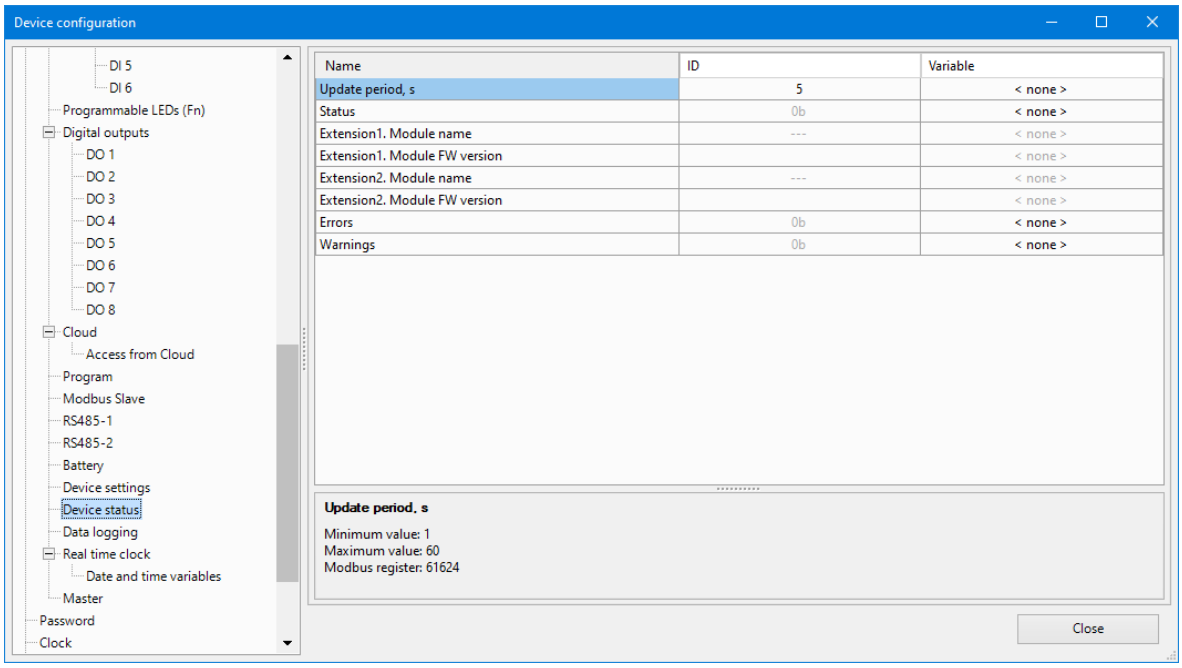



*Fig. 4.23 Device status parameters in ALP*

*Table 4.21 Description of the parameter Errors values*

| Value | Description | Note |
|---|---|---|
| 0 | No base clock frequency | The errors are checked once at the device start |
| 1 | Microcontroller ID is not correct | |
| 2 | Microcontroller freezing caused by a hardware-dependent software error | |
| 3 | The flash-memory SPI bus is not initialized or a not supported flash-memory type installed | |
| 4 | RTC error | |
| 5 | Watchdog timer error | The errors are checked periodically during the device operation |
| 6 | Retain error | |
| 7 | Logic initialization error | |

*Table 4.22 Description of the parameter Warning values*

| Value | Description | Note |
|---|---|---|
| 0 | Battery warning | The errors are checked periodically during the device operation |
| 1 | Ethernet warning | |
| 2 | Safe state warning | |

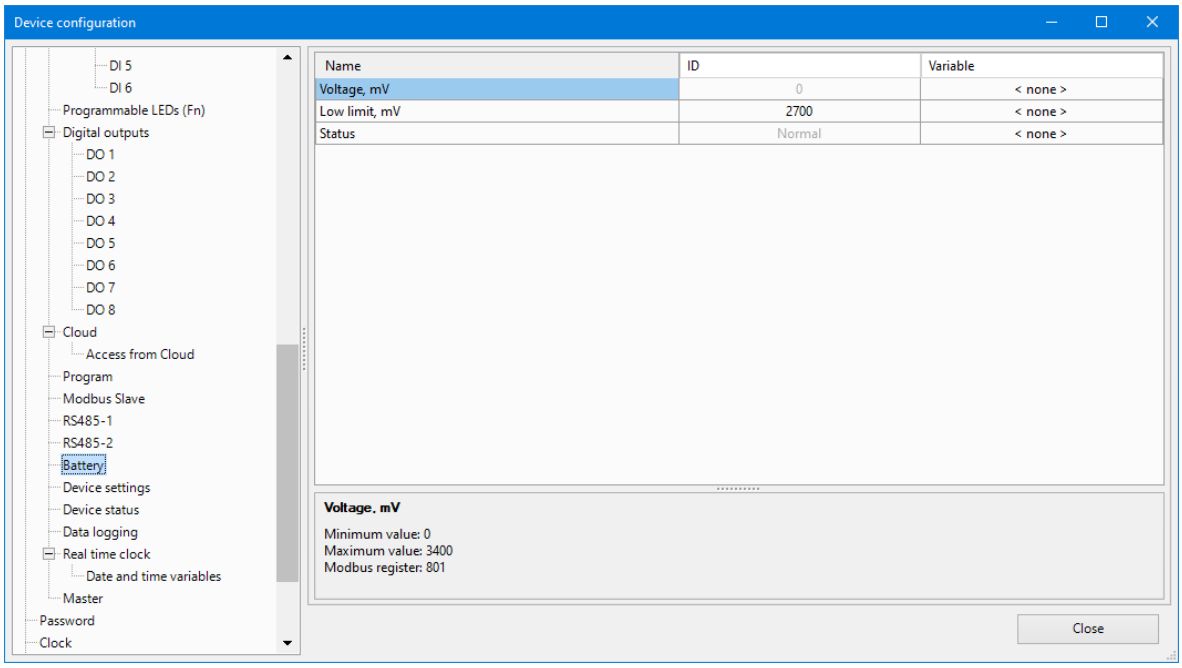



*Fig. 4.24 Battery status parameters*

*Table 4.23 Possible states of battery parameter Status*

| Value | Description |
|---|---|
| 0 | Battery is OK |
| 1 | Battery is discharged |

The status parameters of the connected PRM extension modules are available in ALP.

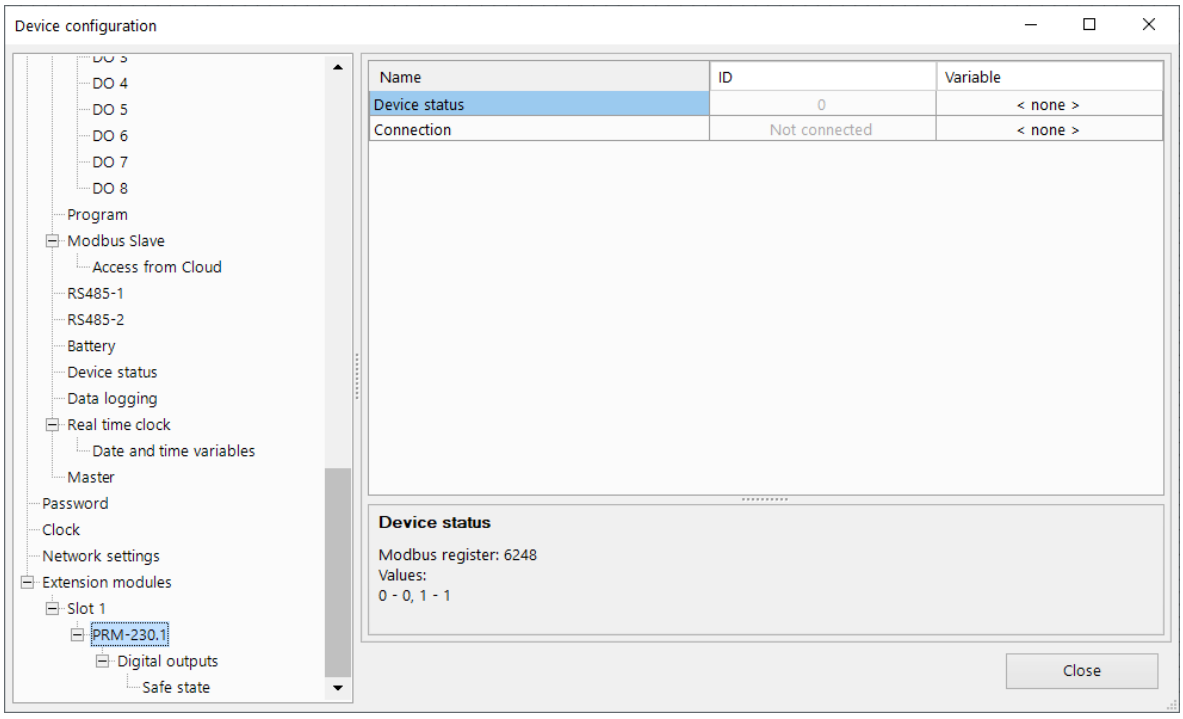



*Fig. 4.25 Status parameters of PRM extension modules*

The value of the parameter ***Device status*** can be read over Modbus. Also a variable can be assigned in the user program to read the parameter value. Description of the parameter possible values is given in the table below.

*Table 4.24 Possible values of PRM extension module parameter Device status*

| Value | Description |
|---|---|
| 0 | No data exchange with the extension module |
| 1 | Data exchange with the extension module is in process |

The value of parameter ***Connection*** can be read over Modbus. Also a variable can be assigned in the user program to read the parameter value. Description of the parameter possible values is given in the table below.

*Table 4.25 Possible values of PRM extension module parameter Connection*

| Status | Value | Description |
|---|---|---|
| Not connected | 0 | No data exchange, the extension module is not connected. |
| Initialization | 1 | Connecting the extension module is in process. |
| Found | 2 | The extension module is found. The type of the extension module installed and the firmware version are not verified. No settings applied. |
| Inappropriate module type | 3 | Communication with the extension module is established but the type of the extension module installed is not appropriate for using with the device. |
| Invalid firmware version | 4 | Communication with the extension module is established but the firmware version of the extension module installed is invalid (out of date). |
| Operation | 5 | Communication with the extension module is established. The type and the firmware version of the extension module installed are correct. The extension module installed is ready for operation with the user software. |

## 4.10 Device configuration

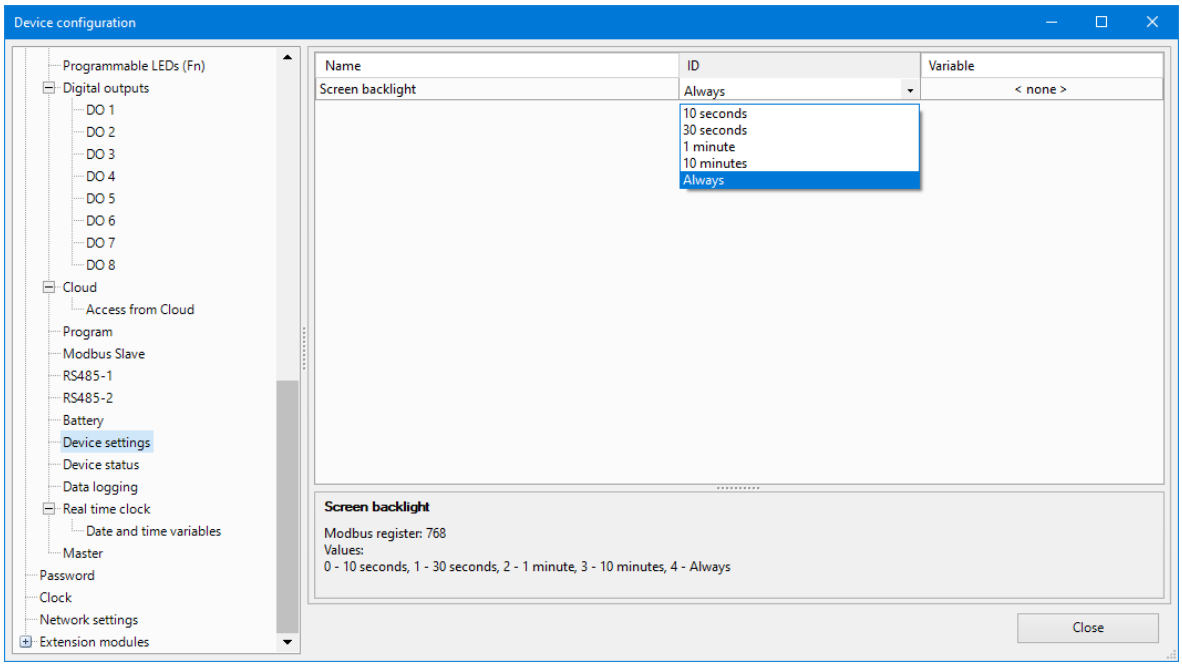



*Fig. 4.26 Device configuration*

| Parameter | Description |
|---|---|
| Backlight | Display backlight duration. Settings:<br>– Always<br>– 10 seconds<br>– 30 seconds<br>– 1 minute<br>– 10 minutes |

### 4.11 Operation with akYtec Cloud

**NOTE**
To connect to the akYtec Cloud server, open port 26502 in the local area network. Specify Google Public DNS (8.8.8.8) as a DNS server in the settings of the connected devices.

The device must be powered externally and connected to the Internet via the Ethernet interface. To connect to akYtec Cloud you should:

1. Connect the device to the PC and create a project using network variables.
2. Set a password to access the device (see *Section 4.7.1*).
3. In the Device configuration window, allow access to akYtec Cloud.

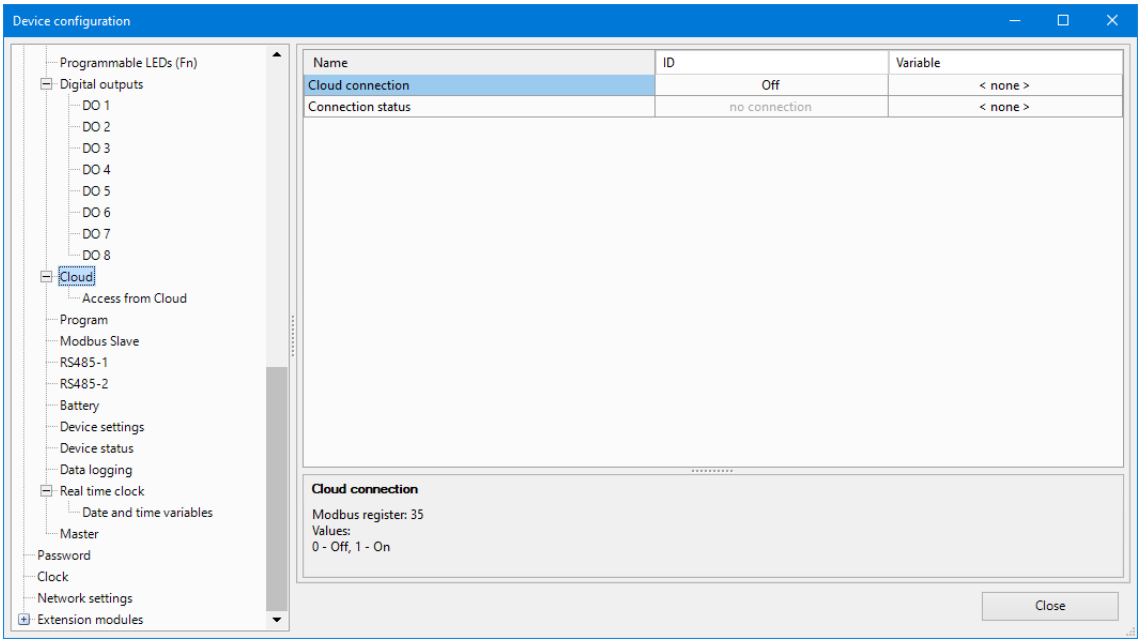



*Fig. 4.27 Device configuration window*

**NOTE**
The value of the **Cloud connection** status parameter can be read via Modbus or a user program variable can be linked. The parameter description is given in the table below.

*Table 4.26 Possible states of the Cloud connection parameter*

| State | Value | Description |
|---|---|---|
| no connection | 0 | Cloud sharing is disabled |
| identification | 1 | Establishing connection to Cloud |
| normal operation | 2 | Cloud controls the device, no errors |
| network error | 3 | The device cannot establish a connection to the Cloud server |
| no password | 4 | The password for the device is not set |

4. Allow remote access to Modbus registers.

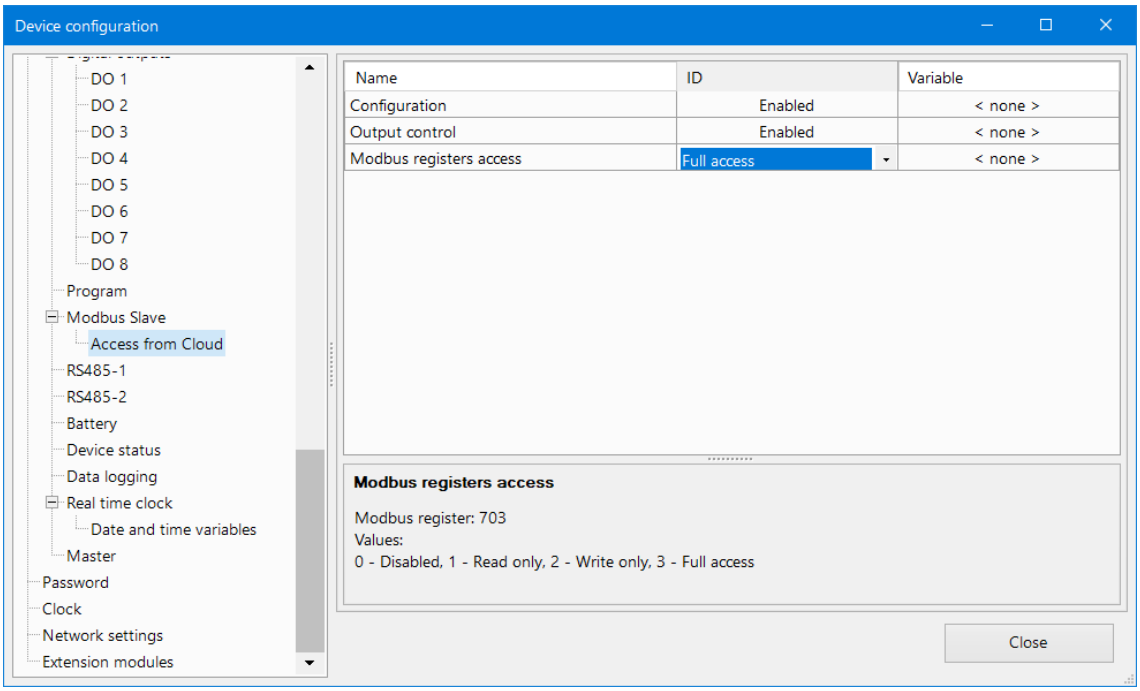



*Fig. 4.28 Modbus registers access*

**NOTE**
It is also possible to configure the device for akYtec Cloud operation using akYtec ToolPro. However, network variables will not be available if the program is not loaded.

5. Load the program into the device **(Device → Load the program into the device**).

**NOTE**
Make sure the **Run/Stop** switch is in the **ON** position.

6. Go to the main page of the *Cloud* service. If there is no registration, then you need to go through the registration procedure.
7. Go to the **Administration** tab, open the **Devices** tab and click the **Add device** button (Add).
8. A window will appear with a choice of the device type.

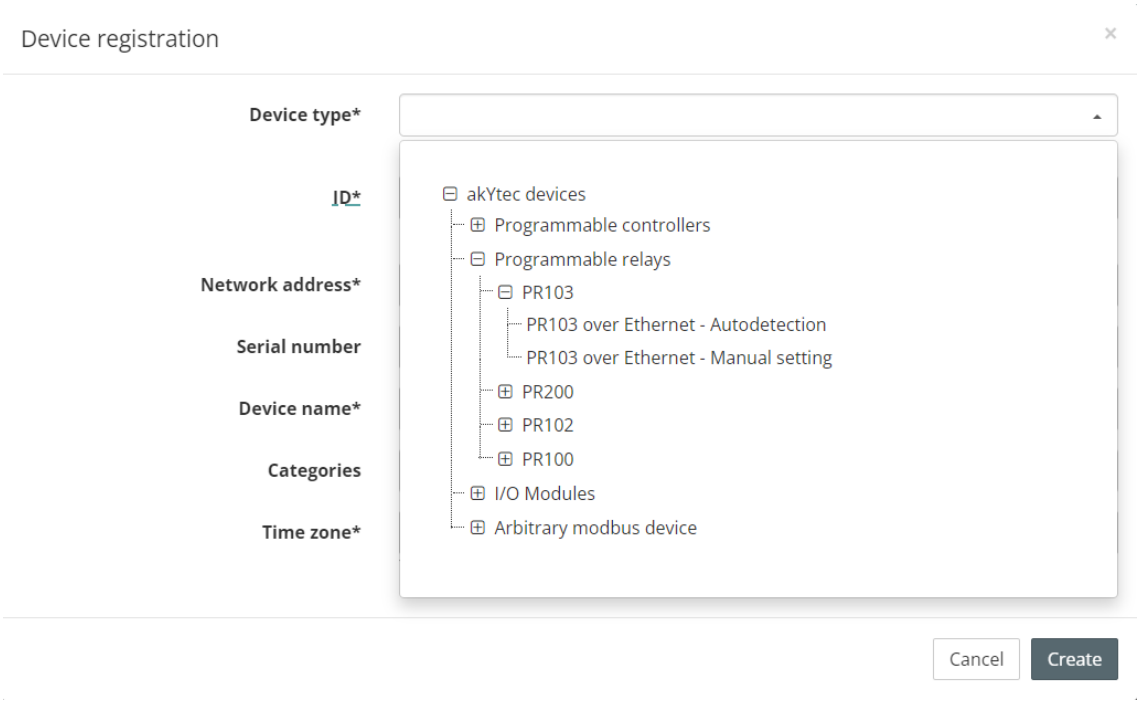



*Fig. 4.29 Selecting the device type*

Option **PR205 via Ethernet - Autodetection**:

a. In the resulting window, in the fields:
   - **ID**: enter the serial number of the device;
   - **Network address**: leave unchanged address **1**;
   - **Device name**: enter the name of the device;
   - **Time zone**: specify the time zone in which the device is located.

b. Click the "Add" button. The basic settings interface of the device will open .

c. Enter the password for the device. If necessary, you can change other settings (for example, the polling period).

d. Click the "Save" button to apply the new settings.

e. Cloud will connect to the device and read all parameters from it.

Option **PR205 via Ethernet - Manual setup**:

a. In the resulting window, in the fields:
   - **ID**: enter the serial number of the device;
   - **Network address**: leave unchanged address 1;
   - **Device name**: enter the name of the device;
   - **Time zone**: specify the time zone in which the device is located.

b. Click the "Add" button. The general settings interface of the device will open .

c. Click the **Create** button to apply the new settings.

d. In the Parameters/Parameter settings tab, add the network parameters of the device in the following way:
   - as a *.json file, if you use the **Device Export** extension in akYtec Cloud ALP. To add parameters, click on the **Import** drop-down list and select the **Load from JSON** option. In the menu that opens, select the previously created file in *.json format and click the **Load parameters** button.

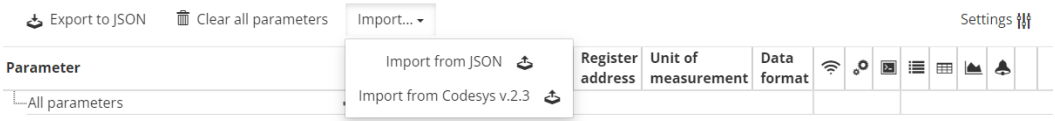



*Fig. 4.30 Import parameters*

Option **PR205 via RS485 - Through a gateway**:

a. In the resulting window, in the fields:
   - **ID**: enter the factory number of the gateway;
   - **Network address**: set the network address in accordance with the address of the port to which the gateway is connected;
   - **Device name**: enter the name of the device;
   - **Time zone**: specify the time zone in which the device is located.

b. Click the "Add" button. The Cloud sharing setup window will appear.

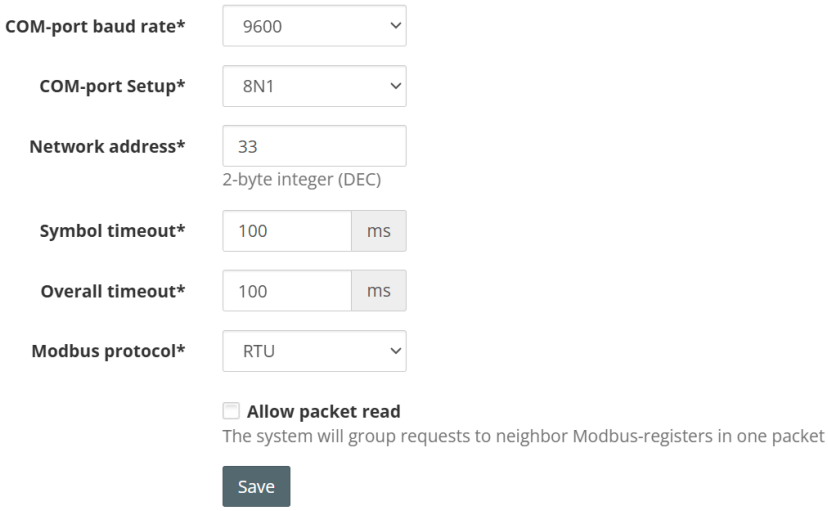



*Fig. 4.31 Cloud Sharing Settings*

Fields:
- **COM-port Baud rate**
- **COM-port**: set the **COM port** according to the address of the port to which the gateway is connected;
- **Modbus protocol**: enter the name of the device;

The fields must correspond to the similar settings of the port to which the gateway is connected.

c. Click the **Create** button to apply the new settings.

d. In the **Parameters/Parameter settings** tab, add the network parameters of the device in the following way:
   - as a *.json file, if you use the **Device Export** extension in akYtec Cloud ALP. To add parameters, click on the **Import** drop-down list and select the **Load from JSON** option. In the menu that opens, select the previously created file in *.json format and click the **Load parameters** button.

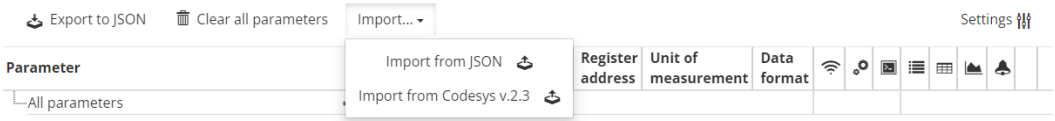



*Fig. 4.32 Import parameters*

If all settings were correct, then the **General Data** tab will display data from the device. If the device is not polled from Cloud, then you should check the network settings, the connection status to Cloud.


*Fig. 4.33 Example display of connection to Cloud*

If the values are not written to the device from Cloud, you should check if the checkbox in the **Manageable Parameter** column is checked:

*Fig. 4.34 Controlled parameter*

### 4.12 Real-time clock

For setting the clock, open the ***Clock*** node in the ALP configuration dialog.

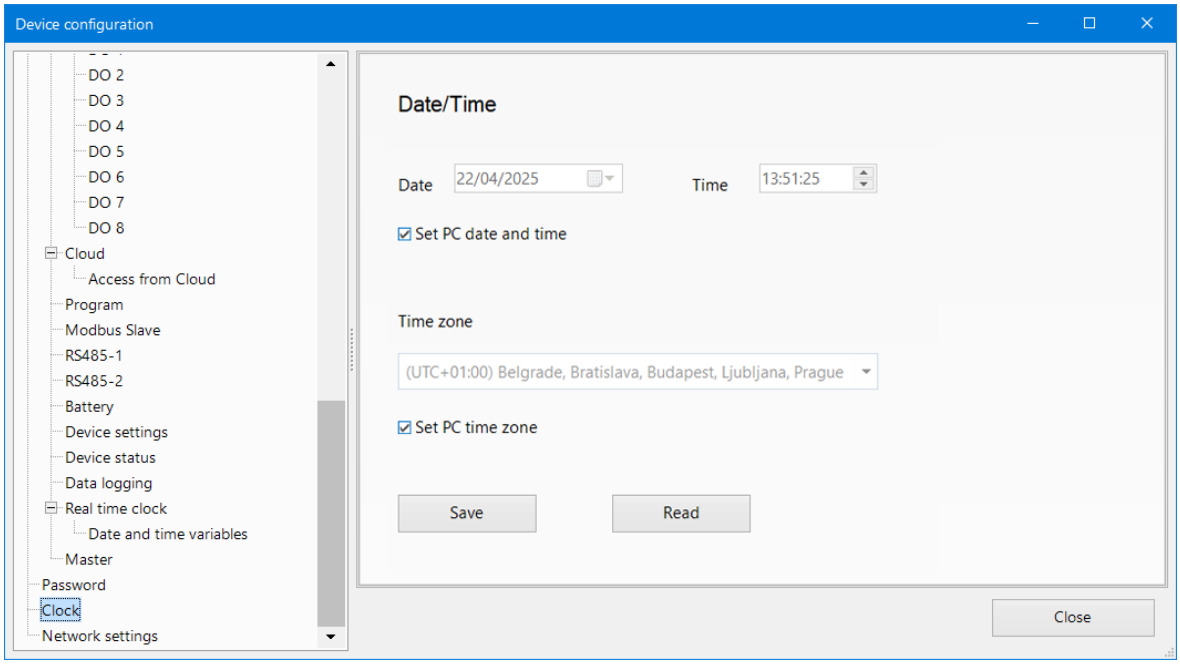



*Fig. 4.35 Date and time settings*

*Table 4.27 Clock settings*

| Parameter | Description |
|---|---|
| ***Set PC date and time*** | As the box is checked, the PC date and time will be recorded to the device after pressing the ***Save*** button. |
| ***Set PC time zone*** | As the box is checked, the PC time zone value will be recorded to the device after pressing the ***Save*** button. |

The user program variables can be assigned to the settings of the RTC parameters in order to change the RTC settings in the user program. Proceed to the ***Real time clock*** tree to assign the user program variables.

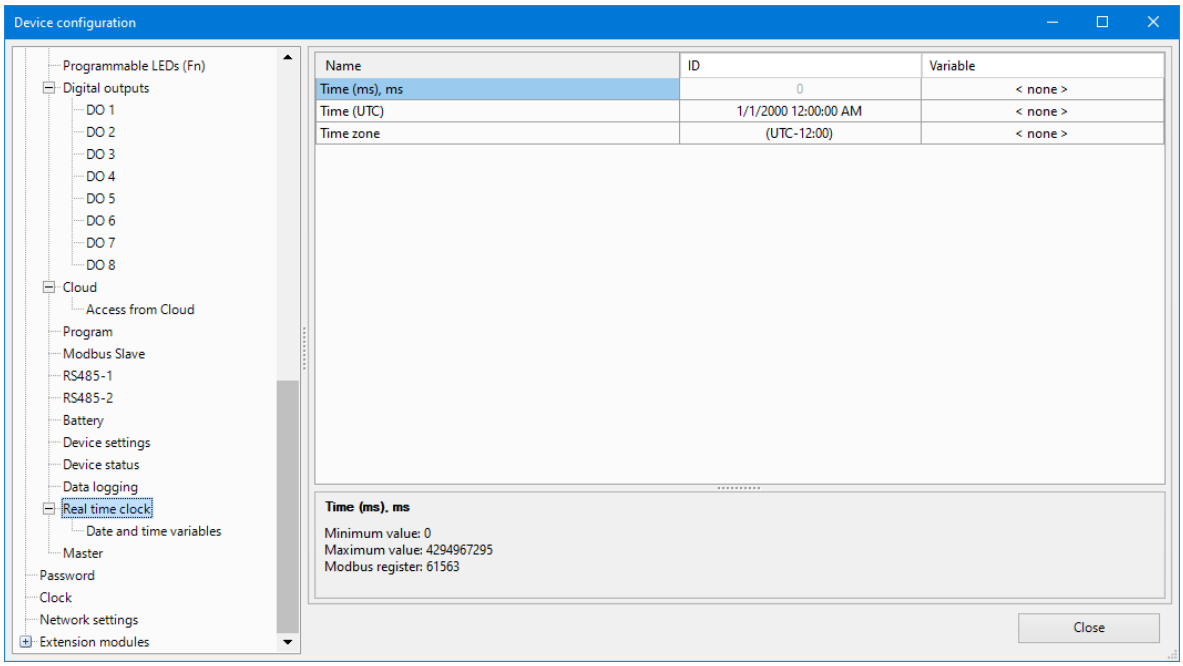



*Fig. 4.36 Assignment of variables to the RTC settings*

*Table 4.28 RTC setting parameters*

| Parameter | Description |
|---|---|
| ***Time (ms), ms*** | The device operation time starting from the powering the device from either USB or external power supply. |
| ***Time (UTC)*** | The real time counted as UTC in seconds, starting from 0:00 01.01.2000 |
| ***Time zone*** | The parameter is for setting the appropriate time zone of the device location |

The user program variables can be assigned to the RTC date and time setting values. In the same ALP dialog tree, proceed to node ***Date and time variables*** to assign the user program variables.

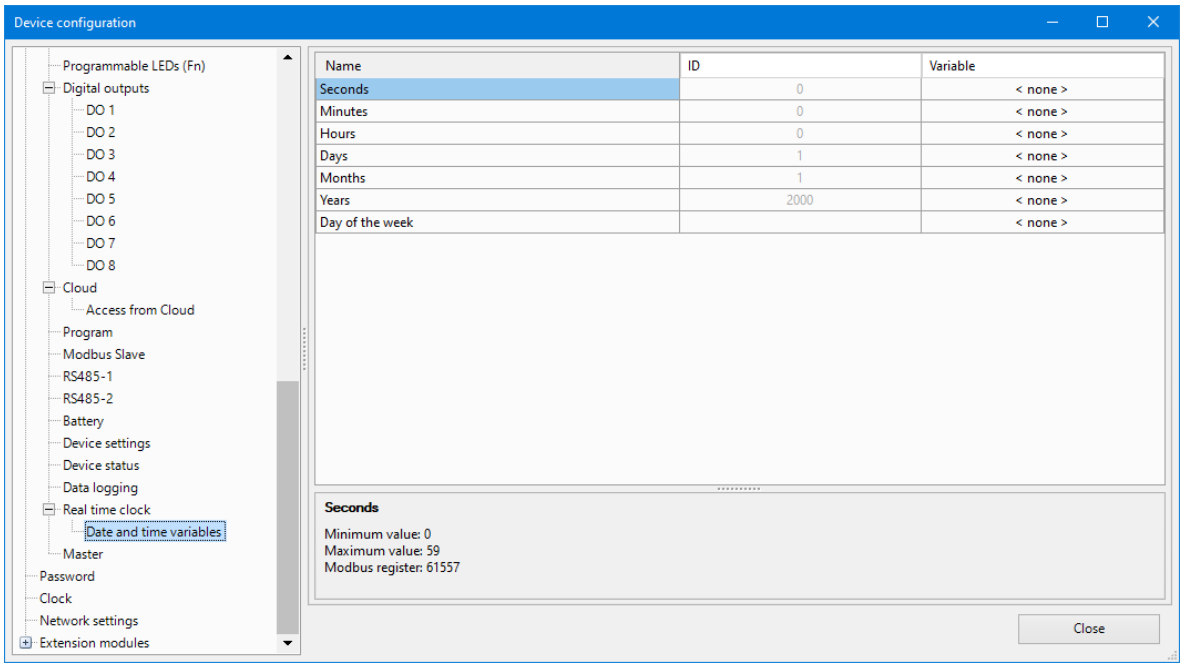



*Fig. 4.37 Assignment of variables to the RTC date and time setting values*

### 4.13 Factory settings restoration

**CAUTION**
***After restoring the factory settings:***
- ***user program is deleted***
- ***password is deleted***
- ***all parameters (except for the Ethernet IP addresses) are reset to factory values***

To restore the factory settings:

1. Power on the device.
2. Holding on to the ribbed area, open the front panel.
3. Using a thin tool, press and hold the service button for at least 12 seconds.
4. Release the button. The device will restart and all parameters will be reset to factory values.
5. Power off the device.

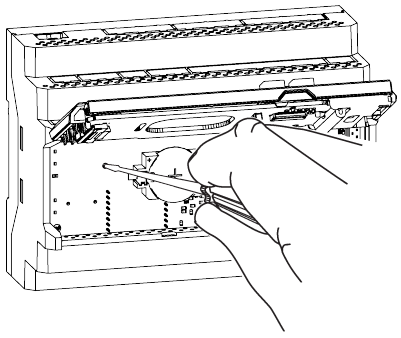


*Fig. 4.38 Factory settings restoration*

When switched on again, the device starts with the factory settings.

# 5 Installation

## 5.1 Montage

The safety measures specified in the *Section 1.4* must be observed during the device montage
The device is to be mounted in enclosures, cabinets, e. t. c. with protection of the device from dust, moisture, and foreign objects.

**NOTICE**
***Configure and program the device prior to montage and wiring.***

**CAUTION**
***Do not use the device power terminals for powering any other equipment !***

The relay is designed for DIN rail mounting. The operating conditions (see *Section 3.2*) should be taken into account when choosing the installation site.
Follow the steps below for mounting the device on the DIN rail:

1. Prepare the place on the DIN rail where the device should be mounted. Take into account the device overall dimensions (see *Fig. 5.1*).

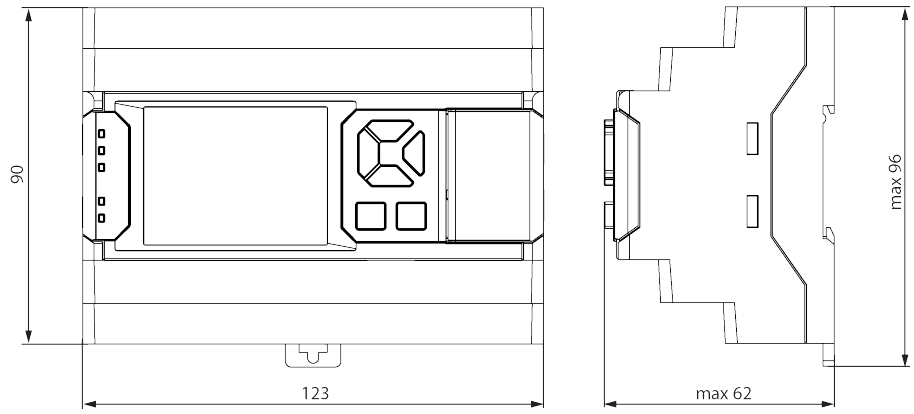



*Fig. 5.1 Dimensions*

2. Insert a screwdriver into the eyelet of the slide interlock and then pull it down to loosen the slide interlock (see *Fig. 5.2*, 1). Position the device onto the DIN rail.
3. Press the device firmly against the DIN rail in the direction of arrows 1 and 2 (see *Fig. 5.2*, 2). Using a screwdriver, pull up the slide interlock to lock it.
4. Connect the device to the external equipment using the removable terminal blocks (supplied with the device).

In order to remove the device from the DIN rail, follow the steps below:

1. Unplug the removable terminal blocks from the device mating connectors (see *Section 5.2*).
2. Insert a screwdriver into the eyelet of the slide interlock.
3. Pull the slide interlock down to unlock it, then remove the device from the DIN rail.

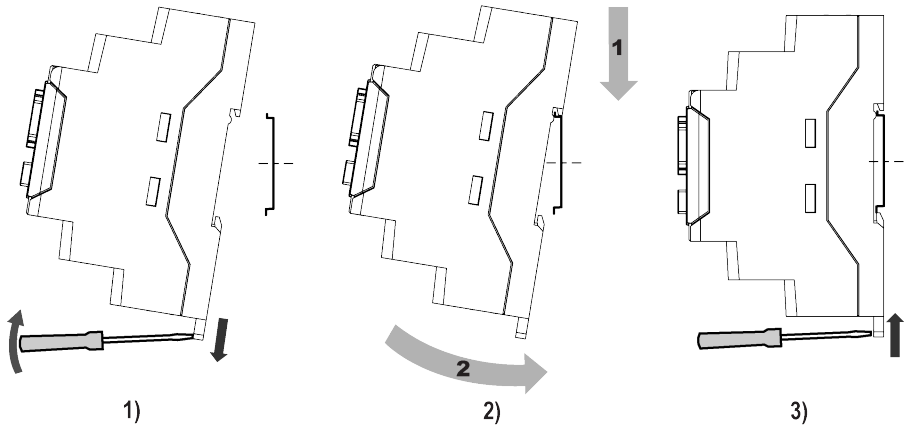



*Fig. 5.2 DIN rail mounting*

### 5.2 Quick replacement

Relay is equipped with plug-in terminal blocks which enable quick replacement of the device without disconnecting the existing wiring.
For the device quick replacement, follow the steps below:

1. Power off all connected lines including power supply.
2. Using a screwdriver or a similar tool, unplug the terminal blocks with existing wiring connected (see *Fig. 5.3*).
3. Remove the device from the DIN rail and install another PR205 of the same modification (with the terminal blocks unplugged).
4. Plug the terminal blocks with existing wiring into mating connectors of the PR205 installed.

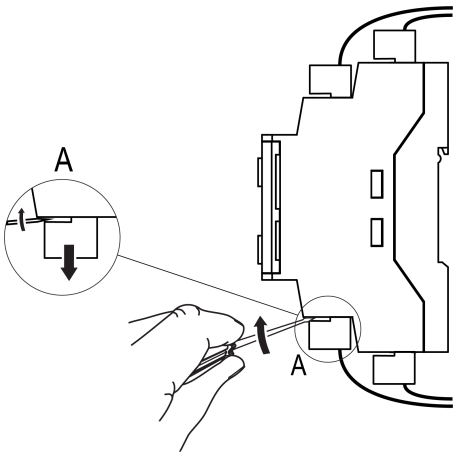



*Fig. 5.3 Quick replacement*

# 6 Wiring

## 6.1 Wiring recommendations

**CAUTION**
***Do not use the device power terminals for powering any other equipment !***

In order to ensure reliable electrical connections, use stranded copper wire cables. Twist and tin the wire conductors or terminate them using cable ferrules. Strip off the wire insulation, so that the exposed wire ends would not extend beyond the terminal blocks when connected. The wire cross-section must not exceed 2.5 $mm^2$.
Connect the microUSB programming connector of the device to PC USB port for transferring the user program to the device.

**CAUTION**
***The device must be powered off before connecting to peripheral equipment or PC. Switch on the power supply only after the wiring of the device is completed.***

Analog inputs and outputs, as well as transistor outputs and the USB interface of the device are not galvanically isolated. In order to prevent the device damaging, ensure the equipotential grounding of the equipment which are connected to these inputs and outputs. If it is not possible, do not connect the external equipment to the analog inputs and outputs, the transistor outputs and the USB interface simultaneously. Disconnect external cables from the analog inputs and outputs, as well as from the transistor outputs, when programming the device over the USB interface or use galvanic isolators (e. g. USB isolators, a battery-powered laptop, etc.).

**NOTICE**
***It is prohibited to tie together the common grounds of the device inputs and outputs as well as connecting them to the earth ground of the enclosure where the device is installed.***

**NOTICE**
***In order to avoid a risk of high voltage at the device connectors, use power supplies with the reinforced isolation for powering sensors and other peripheral equipment connected to the analog and digital inputs of the device.***

**CAUTION**
***It is prohibited to power the device and the connected sensors from the same power supply!***

## 6.2 Interference suppression

The device operation may be affected by electromagnetic interference (EMI) caused by external electromagnetic fields. EMI is induced in the device circuitry and the connected cables.
The following measures should be provided for EMI suppression:

– Use shielded signal cables. The cable shields must be electrically isolated from the peripheral equipment all along their routes. The shields of signal cables must be connected to an earthing terminal.
– Install the device in a metal enclosure with no any power equipment inside. The enclosure must be earthed.

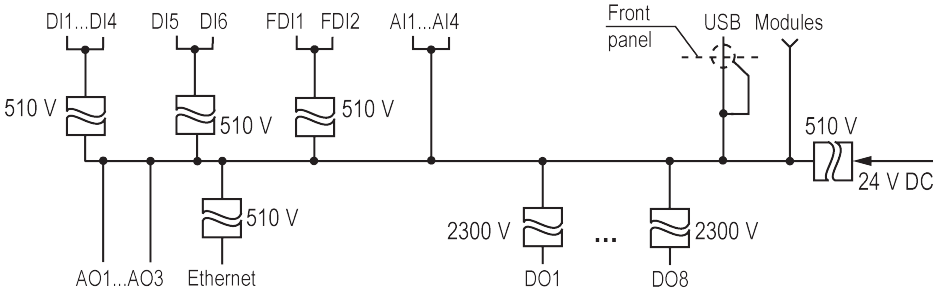
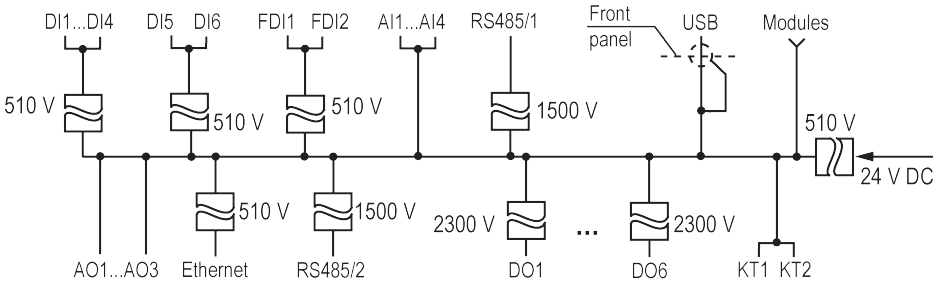
In order to reduce the interference effect, use the program debouncing filters which can be set individually for each of the device inputs. The program debouncing filters are available for the following device inputs:

– Analog inputs.
– 24 V digital inputs.

**NOTE**
***Increasing the debouncing time results in a slower device response while processing fast changing input signals.***

### 6.3 Galvanic isolation

*Table 6.1 PR205 galvanic isolation*

| PR205 modification | Galvanic isolation diagram |
|---|---|
| PR205.24.1.2 |  |
| PR205.24.5.2 |  |

### 6.4 Sensors wiring

#### 6.4.1 Digital inputs – switch contacts wiring

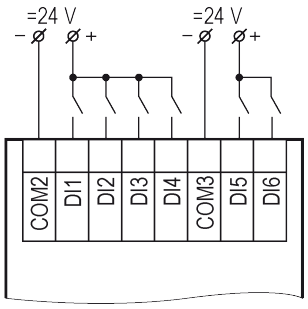



*Fig. 6.1 DI – 24 V switch contacts wiring*

#### 6.4.2 Fast digital inputs – push-pull, NPN and PNP output sensors wiring

Each fast DI has an RC filter with R = 750 Ω and C = 470 pF. The minimum level of the LOW-voltage maximum of input $U_{DI\,lmin}$ is 6.5 V. The maximum level of the HIGH-voltage minimum of input $U_{DI\,hmax}$ is 8.55 V.

**Push-pull output sensor**

The push-pull output of the sensor is wired directly to the device digital input.

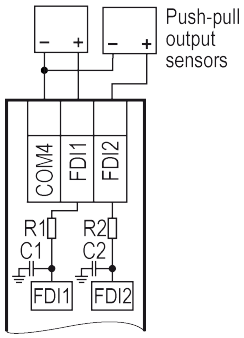



*Fig. 6.2 Push-pull output sensor wiring*

**NPN output sensor**

An additional resistor ($R_D$) must be used when connecting the sensor NPN transistor output to the device fast digital input (see the figure below).

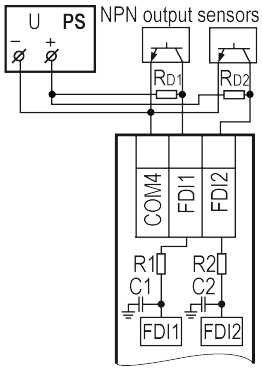



*Fig. 6.3 DI – NPN output sensor wiring*

The $R_D$ nominal value depends on:
• supply voltage $U_p$;
• the time when the sensor transistor is closed ($t_{cl\_VT}$);
• the required minimum of HIGH voltage.
The minimum voltage ($U_{p\ min}$) taken by the device as HIGH voltage is calculated by the formula:

$$U_{p.min} = U_{DIhmax} + I_{nomDI} \cdot (R1 + R_D) \quad (6.1)$$

Where $I_{nom\ DI}$ = 2.72 mA – nominal current of the FDI. If $U_p$ is below the rated value ($U_{p\ min}$), the device will consider this voltage as HIGH.
Since an RC filter is installed at the device digital input, time $t_{cl\_VT}$ and voltage $U_p$ are the critical factors for capacitance C1 to have time to get charged to the maximum level of the HIGH-voltage minimum of the input ($U_{DI\ h\ max}$).
If the supply voltage ($U_p$) is too low and the resistance ($R_D$) is too high, there may not be enough time for capacitance C1 to charge to 8.55 V when the sensor transistor is closed.
The voltage to which C1 is charged is calculated by the formula:

$$U_t = U_p \cdot (1 - e^{-\frac{1}{T}}) \quad (6.2)$$

where:
$U_t$ – the voltage to which capacitor C1 is charged;
t – charge time.
T is calculated as follows:

$$T = (R1 + R_D) \cdot C1 \tag{6.3}$$

$U_t$ must be above 8.55 V and remain like this for at least 5 µs for a fast digital input. The $R_D$ maximum value, at which the voltage ($U_{DI}$) reaches 8.55 V, is calculated by the formula:

$$R_D \leq \frac{-t}{C1 \cdot ln\left(1 - \frac{U_t}{U_p}\right)} - R1 \tag{6.4}$$

Time period, when the sensor transistor is closed for a fast digital input, should be of $U_t$ + 5 µs. It is necessary to choose a sensor with the NPN transistor output taking into account the rated current and power dissipation, and to choose a resistor ($R_D$) taking into account power dissipation.
The second limitation of the $R_D$ nominal value follows from *formula* 6.1:

$$R_D \leq \frac{U_p - U_{DIhmax} - I_{nomDI} \cdot R_1}{I_{nomDI}} \tag{6.5}$$

***Example:***
Given the NPN sensor is connected to the device and a power supply of 12 V,
assume $t_{cl\_VT}$ = 5 µs.
For reliable operation, the voltage ($U_{DI}$) must reach 8.55 V in 3 µs and must not fall below this value for at least 2 µs.
The $R_D$ nominal value must meet the conditions of *formula* 6.4 and *formula* 6.5 *(RD≤)*:

$$R_D \leq \frac{-5 \cdot 10^{-6}}{470 \cdot 10^{-12} \cdot ln\left(1 - \frac{8{,}55}{12}\right)} - 750 \leq 7784\,Ohm$$

$$R_D \leq \frac{12V - 8.55V - 0.00272\,A \cdot 750\,Ohm}{0.00272\,A} \leq 518\,Ohm$$

Thus, the $R_D$ nominal value should not exceed 518 Ω.
Some of the $R_D$ nominal values in relation to the supply voltage and the time period of FDI passive state are shown in the table below.

*Table 6.2 $R_D$ nominal values*

| FDI power supply voltage ($U_p$), V | Time of FDI passive state, ms | Resistance ($R_D$), Ω |
|---|---|---|
| 12 | 0.01 | 500 |
| 24 | | 4500 |

**PNP output sensor**

An additional resistor ($R_D$) must be used when connecting the sensor PNP transistor output to the device fast digital input (see the figure below).

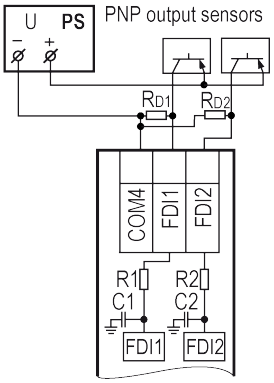



*Fig. 6.4 DI – PNP output sensor wiring*

The $R_D$ nominal value depends on supply voltage $U_p$ and the time when the sensor transistor is closed ($t_{cl\_VT}$).
Since an RC filter is installed at the device digital input, time $t_{cl\_VT}$ is the critical factor for capacitance C1 to have time to get discharged to the minimum level of the LOW-voltage maximum of the input ($U_{DI\ lmin}$).
If the supply voltage ($U_p$) and the resistance ($R_D$) are too high, there may not be enough time for the device input voltage to reach 6.5 V while the sensor transistor is closed.
The voltage to which C1 is discharged is calculated by the formula:

$$U_t = U_p \cdot e^{-\frac{1}{T}} \tag{6.6}$$

where
$U_t$ – the voltage to which capacitor C1 is discharged;
t – discharge time.
T is calculated by the formula:

$$T = (R1 + R_D) \cdot C1 \tag{6.7}$$

$U_t$ must be below 8.55 V and remain like this for at least 5 µs. The $R_D$ maximum value, at which the voltage ($U_{DI}$) reaches 6.5 V, is calculated by the formula:

$$R_D \leq \frac{-t}{C1 \cdot ln\left(\frac{U_t}{U_p}\right)} - R1 \tag{6.8}$$

Time period, when the sensor transistor is closed for a fast digital input, should be not less than $U_t$ + 5 µs.
It is necessary to choose a sensor with the PNP transistor output taking into account the rated current and power dissipation, and to choose a resistor ($R_D$) taking into account power dissipation.
Some of the $R_D$ nominal values in relation to the supply voltage and the time period of FDI passive state are shown in the table below.

*Table 6.3 $R_D$ nominal values and time of FDI passive state*

| FDI power supply voltage ($U_p$), V | Time of FDI passive state, ms | Resistance ($R_D$), kΩ |
|---|---|---|
| 40 | 0.005 | 5 |
| 30 | | 6 |
| 24 | | 7 |

### 6.4.3 Analog inputs – digital output sensors wiring

**NOTICE**
***Before connecting digital output sensors to the analog inputs, make sure that the analog inputs are configured for operation in the digital mode.***

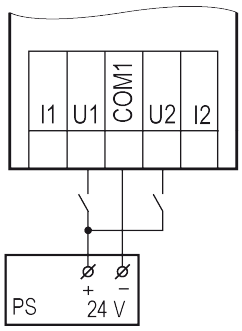



*Fig. 6.5 AI (digital mode) – 24 V switch contacts wiring*

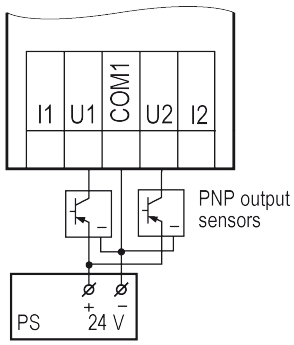



*Fig. 6.6 AI (digital mode) – 3−wire PNP open collector output sensor wiring*

**NOTE**
***The COM1 terminals are internally connected.***

### 6.4.4 Analog inputs – analog output sensors wiring

**NOTICE**
***Before connecting analog output sensors to the analog inputs, make sure that the analog inputs are configured for operation in the analog mode.***

Please refer to *Section 4.2.1* for configuration of the analog inputs.

**CAUTION**
***Before connecting analog output sensors, make sure that the input signal selected in the configuration settings corresponds to the connected one.***

**NOTICE**
***In order to protect the device input circuitry from a possible ESD damage caused by a possible static charge accumulated on ”the device - sensor” cables, connect the cable conductors to a functional earth (FE) terminal for 1–2 seconds before connecting to the device.***

Disconnect the device from the power supply when checking the functionality of the sensors and the connection cables.
In order to prevent the device damage, use the measuring instruments with the maximum supply voltage of 4.5 V for ring-out. Disconnect sensors from the device if the supply voltage of the measuring instruments exceeds 4.5 V.
The parameters of the connection cables used for connecting sensors to the device are given in the table below.

*Table 6.4 Sensor cable requirements*

| Type of the sensor output | Cable length, max. (m) | Total resistance, max. (Ω) | Type of the connection wiring |
|---|---|---|---|
| Resistive output signal | 100 | * see note below | 2-wire (two wires of the same length and cross-section) |
| Unified current output signal (DC) | 100 | 100 | 2-wire |
| Unified voltage output signal (DC) | 100 | 5 | 2-wire |

**NOTE**

**** For the resistive sensor 2-wire connection, the resistance of the wires connected to the sensor is added to the sensor output resistance, thus, introducing a proportional error into the measurement. The additional resistance of the connecting wires must be taken into account when connecting the sensor to the device. Also, the additional resistance of the connecting wires must be considered to correlate with the sensor output operating range.***

***For example, having an RTD Cu 500 (α = 0.00428 °C⁻¹) for measuring the temperature in the range from –50 °C to +200 °C, the RTD output resistance range is from 393.5 Ω to 926 Ω. Therefore, the sensor output operating range is 532.5 Ω. Thus, the additional resistance of the wires of 1 Ω will introduce the error (1×100) / 532.5 = 0.19% into the temperature measurement.***

The 2-wire connection is used for the RTD wiring.

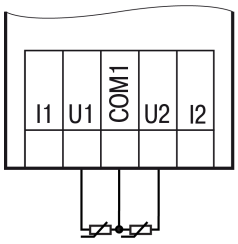



*Fig. 6.7 RTD sensors wiring*

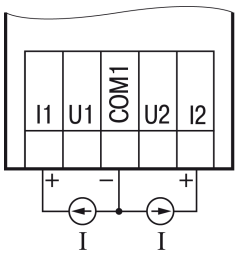



*Fig. 6.8 Current sensors wiring*

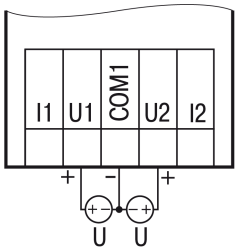



*Fig. 6.9 Voltage sensors wiring*

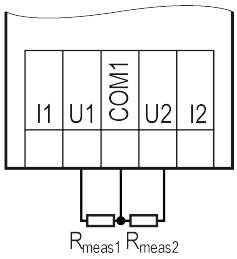



*Fig. 6.10 Resistance sensors wiring*

## 6.5 Output wiring

### 6.5.1 Digital outputs

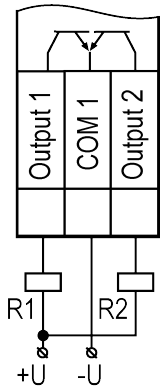



*Fig. 6.11 Transistor outputs wiring*

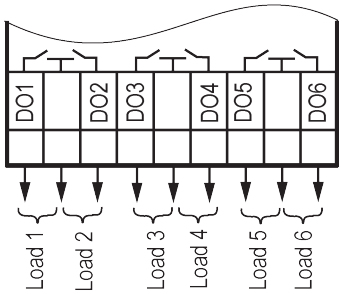



*Fig. 6.12 Relay outputs wiring*

### 6.5.2 Analog outputs

**NOTE**
***Analog outputs require external voltage supply.***

**CAUTION**
***The external supply voltage may not exceed 30 V. Higher voltage can damage the device.***

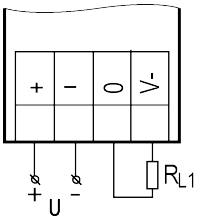



*Fig. 6.13 Current output wiring (4-20 mA output mode)*

Load resistance for the 4-20 mA output signal must not exceed 600 Ω.

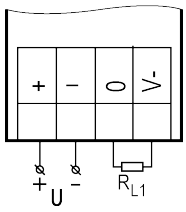



*Fig. 6.14 Voltage output wiring (0-10 V output mode)*

Load resistance for the 0-10 V output signal must not be lower than 600 Ω.

### 6.6 Extension modules connection

**NOTICE**
***The device must be powered off before connecting extension modules.***
***The device and all external equipment must be powered off before connecting the external equipment to the extension modules.***

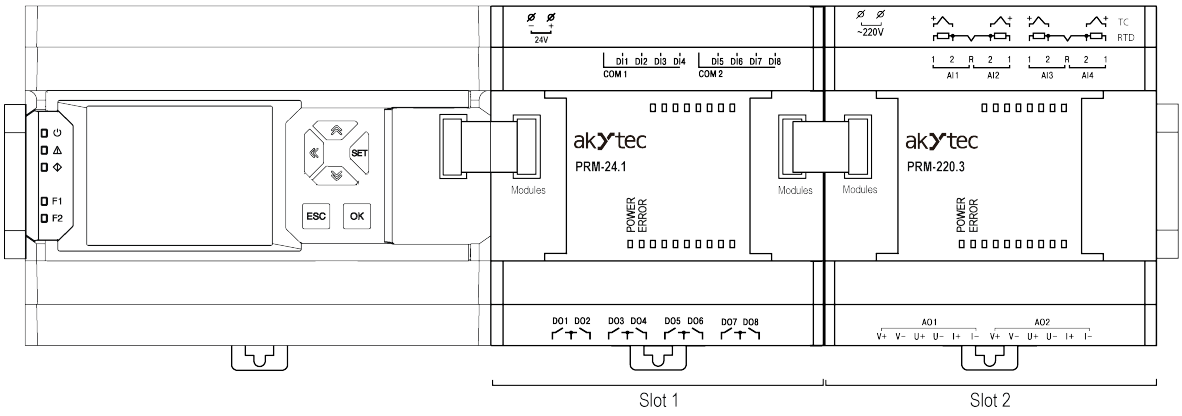



*Fig. 6.15 Extension modules connection*

Maximum two PRM modules can be connected to PR205 in series.
Mount the modules on the DIN rail to the right of PR205 and connect them using the supplied 4.5 cm flat cable.
PRM has two EXT connectors located under the right and left front covers. The connector under the left cover is used to connect the 1st PRM to PR205.
When connected, the flat cable should be placed in a special recess under the cover (see *Fig. 6.16*, arrow 1) to enable the PRM to be pushed close to PR205 (see *Fig. 6.16*, arrow 2).

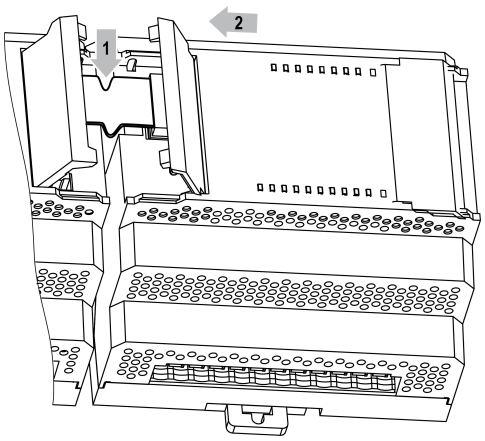



*Fig. 6.16 Extension module flat cable*

Each module has an independent power supply. It is possible to combine the basic device and modules with different supply voltages.
After the first connection to the basic device, the ERROR LED on the module blinks, since there is no data exchange between the module and the basic device. Only when the module is added to the basic device configuration and the project is transferred to the device, the ERROR LED on the module goes out. If that doesn't happen, update the module firmware.

### 6.7 RS485 connection

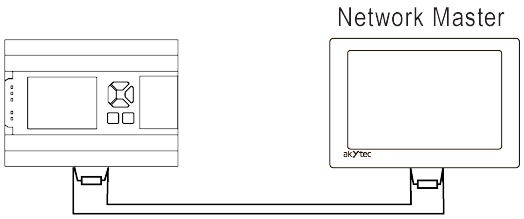



*Fig. 6.17 Typical RS485 connection – PR205 as Slave*

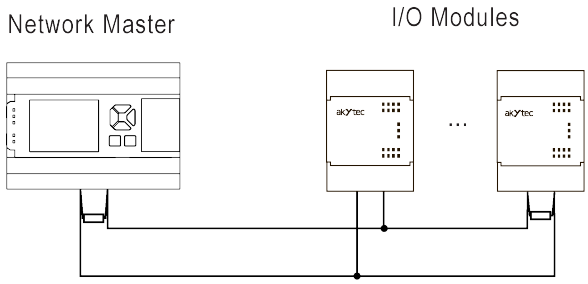



*Fig. 6.18 Typical RS485 connection – PR205 as Master*

### 6.8 Ethernet connection

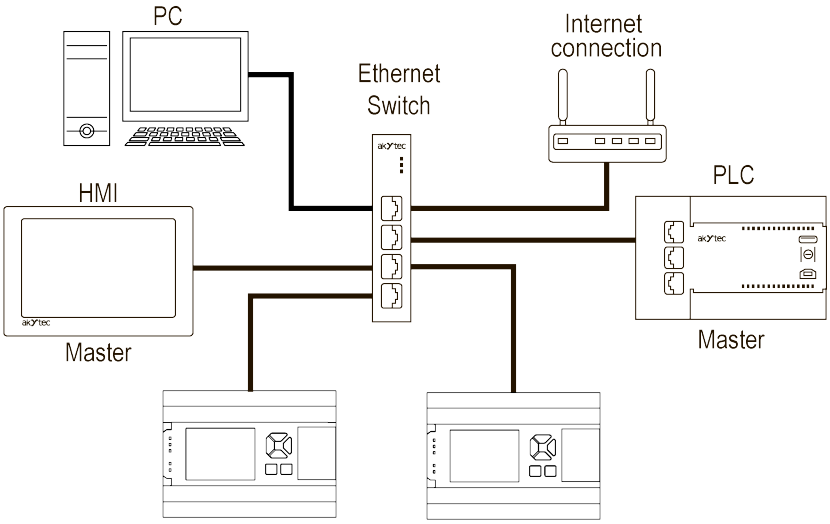



*Fig. 6.19 Star topology*

### 6.9 Connection to akYtec Cloud

**NOTE**
Only a configured device should be connected to akYtec Cloud.

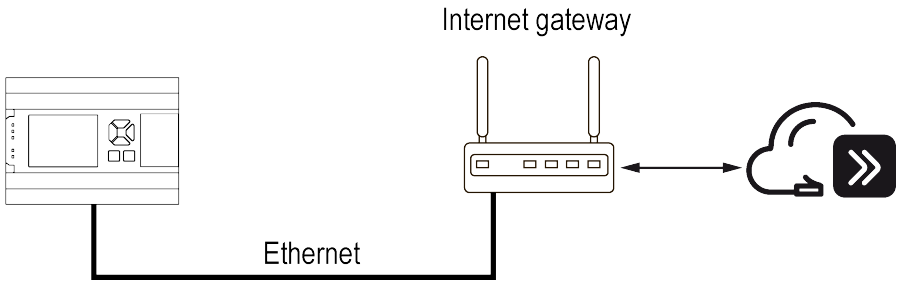



*Fig. 6.20 akYtec Cloud connection*

Use the Ethernet interface for the connection. Communication with akYtec Cloud and device polling via Modbus TCP can be performed simultaneously.
Settings of akYtec Cloud operation are given in *Section 4.11*.

### 6.10 PC connection

Use the microUSB to USB cable for connecting the device to PC over USB interface or use the Ethernet cable for connecting the device to PC over Ethernet.

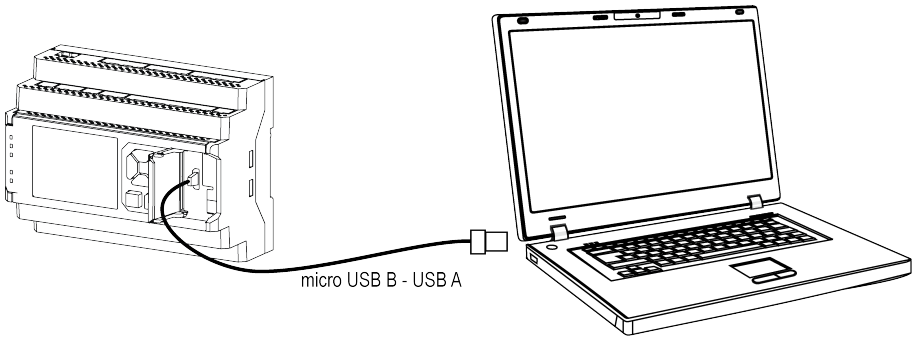



*Fig. 6.21 Connecting PR205 to PC over USB*

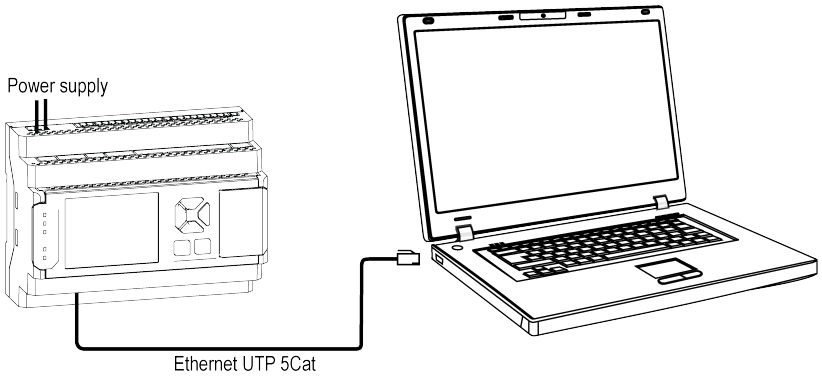



*Fig. 6.22 Connecting PR205 to PC over Ethernet*

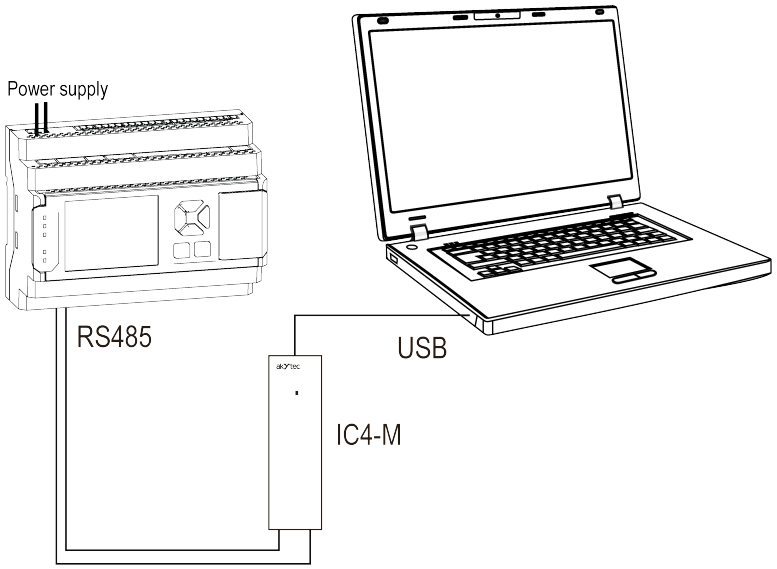



*Fig. 6.23 Connecting PR205 to PC over RS485*

# 7 Operation

## 7.1 Indication, controls and interfaces

Functions of the PR205 buttons can be divided into two groups: basic and programmable. The basic functions are given in the table below. The programmable functions are described in *Section 7.7*.

*Table 7.1 Buttons*

| Button | Description |
|---|---|
| **View mode** | |
| ▲ and ▼ | Screen navigation.<br>Move to the next screen when the selected area is on the border of the current screen. |
| ◄ + ▼ | To the next screen from anywhere on the current screen. |
| ◄ + ▲ | To the previous screen from anywhere on the current screen. |
| OK | Press and hold for 6 seconds to enter the system menu. |
| ESC | Press and hold for 6 seconds to exit the system menu. |
| **Edit mode** | |
| SEL | Enter the edit mode on the current screen.<br>When pressed, the first editable element on the screen becomes available for editing and starts flashing.<br>Apply a value and move to the next parameter. |
| ▲ and ▼ | Change the parameter value.<br>Press and hold to accelerate the value change. |
| ◄ | Move to a higher level. When moving to the maximum level, it jumps to the lowest one. |
| ◄ + ▼ | Move to a lower level. |
| ◄ + ▲ | Move to a higher level. |
| ESC | Exit the edit mode without saving the edited value. |
| OK | Exit the edit mode and save the edited value. |

*Table 7.2 Functional assignment of the LED indicators*

| LED | Color | Status | Description |
|---|---|---|---|
| ⏻ | green | ON | Power on terminals 1 and 2 |
| ⚠ | red | ON | Failure in the device. Contact the service center |
| | | Flashing | |
| **F1** | green | — | To be defined by user's program |
| **F2** | red | | |
| ◈ | red<br>green | OFF<br>Flashing | The RUN/STOP switch is in the **STOP** position. The device operates in the I/O mode (see *Section 7.9.3*) |
| | red<br>green | ON<br>OFF | No power on terminals 1 and 2.<br>Powered by USB. |
| | red<br>green | OFF<br>OFF | User's program is not loaded. PR205 is not configured |
| | red<br>green | OFF<br>ON | The RUN/STOP switch is in the **RUN** position. User's program is running |
| | red<br>green | Flashing*<br>ON | The RUN/STOP switch is in the **RUN** position |

| LED | Color | Status | Description |
|---|---|---|---|
| | red<br>green | OFF<br>Flashing* | The RUN/STOP switch is in the **STOP** position |
| | red<br>green | Flashing*<br>OFF | PR205 is not configured |
| | red<br>green | Flashing with a period<br>ON | Error |
| | red<br>green | ON<br>ON | Expecting for the start of the firmware boot |
| | red<br>green | Flashing<br>Flashing | Boot is in progress |
| * Flashing together with ⚠ means that the RTC battery is discharged | | | |

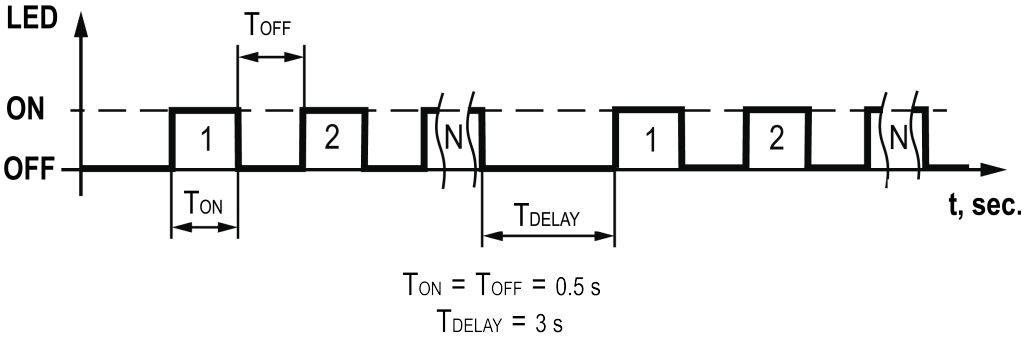



*Fig. 7.1 Fatal and non-fatal errors indication diagram*

*Table 7.3 Fatal errors indication*

| Number of LED pulses (N) | Description |
|---|---|
| 1 | PR205 built-in microcontroller error |
| 2 | |
| 3 | |
| 4 | Internal bus initialization failure |
| 5 | Built-in RTC pulse generator failure |
| 6 | Cyclic reboot caused by incorrect user's program |
| 7 | Retain error |
| 8 | Memory size is not sufficient for user's program |
| 9 | PCB version installed is not correct |

*Table 7.4 Non-fatal errors indication*

| Number of LED pulses (N) | Description |
|---|---|
| 1 | RTC and retain memory battery discharged |
| 3 | Ethernet interface failure |

For detailed information on PR205 errors, see *Section 7.9.2*.

*Table 7.5 RUN/STOP switch*

| Position | Function |
|---|---|
| RUN | When the device is powered on, the user program starts. |
| STOP | When the device is powered on, it functions as an I/O module. The user program is stopped. |

*Table 7.6 Service button*

| Duration of pressing | Function |
|---|---|
| 2 s | IP address assignment (see *Section 4.8.2.1*) |
| 12 s | Restoration of the device factory settings (see *Section 4.13*) |

The Ethernet port is equipped with the two status LEDs. The LEDs indication modes are described in the figure and the table below.

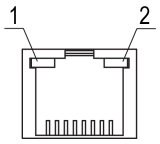



*Fig. 7.2 Location of the Ethernet LEDs: 1 – Amber, 2 – Green*

*Table 7.7 Functional assignment of the Ethernet port LEDs*

| LED | Status | Description |
|---|---|---|
| Amber | OFF | Data transfer rate is 10 Mbit/s |
| | ON* | Data transfer rate is 100 Mbit/s |
| **NOTE**<br>**** With the Ethernet cable disconnected it indicates that the Ethernet port is powered.*** | | |
| Green | OFF | No connection established |
| | ON | Connection established |
| | Flashing | Data transfer is in process |

### 7.2 LCD functions

The LCD is designed to display graphical and textual information. You can create controls in the user-defined forms. In ALP, it is possible to create several forms displaying the state of a managed object including controls. The forms are interconnected by jumps.
The following types of graphic objects are supported:

– indicator,
– progress bar,
– dynamic text,
– label,
– float/int,

etc.
The full list of supported items is presented in the *akYtec ALP User Manual*.
The LCD can display a set of characters within the Windows-1251 encoding. Additional supported symbols and characters see in the table below:

*Table 7.8 Additional characters*

| Description | Symbol |
|---|---|
| Degree | ° |
| Squared | ² |
| Cubed | ³ |
| Alpha lowercase* | α |
| Beta lowercase* | β |
| Gamma lowercase* | γ |
| Delta uppercase* | Δ |
| Eta lowercase* | η |
| Sigma uppercase* | Σ |
| Tau lowercase* | τ |
| Phi lowercase* | φ |
| Psi lowercase* | ψ |
| Omega uppercase* | Ω |
| Number | № |

**NOTE**
* Supported in ALP 2.8.361 and later versions.

Display elements can be editable and non-editable, depending on the type of element and its properties set using ALP.

### 7.3 Working with the menu

The device has a user menu and a system menu.
The user menu is created in ALP with the help of **Display Manager**. To specify “jumps”, use buttons or change a variable. The system menu is always present in the device, even if there is no user program written to it. The principles of working with the user menu and the system menu are similar.
It is possible to work with the menu in the following modes:

– view mode;
– edit mode.

In the **View** mode, you can view the device parameters or the user menu.
In the **Edit** mode, you can edit the device parameters in the system menu or the user program from the front panel without stopping the device. When you re-enter the Edit mode, the last edited element is selected.

### 7.4 System menu

The device system menu is available from any program screen, when there is no user program in the device memory. Press OK button for 6 seconds to access the menu. Press ESC button for 6 seconds to exit the menu.
The menu structure is presented in the figure below.

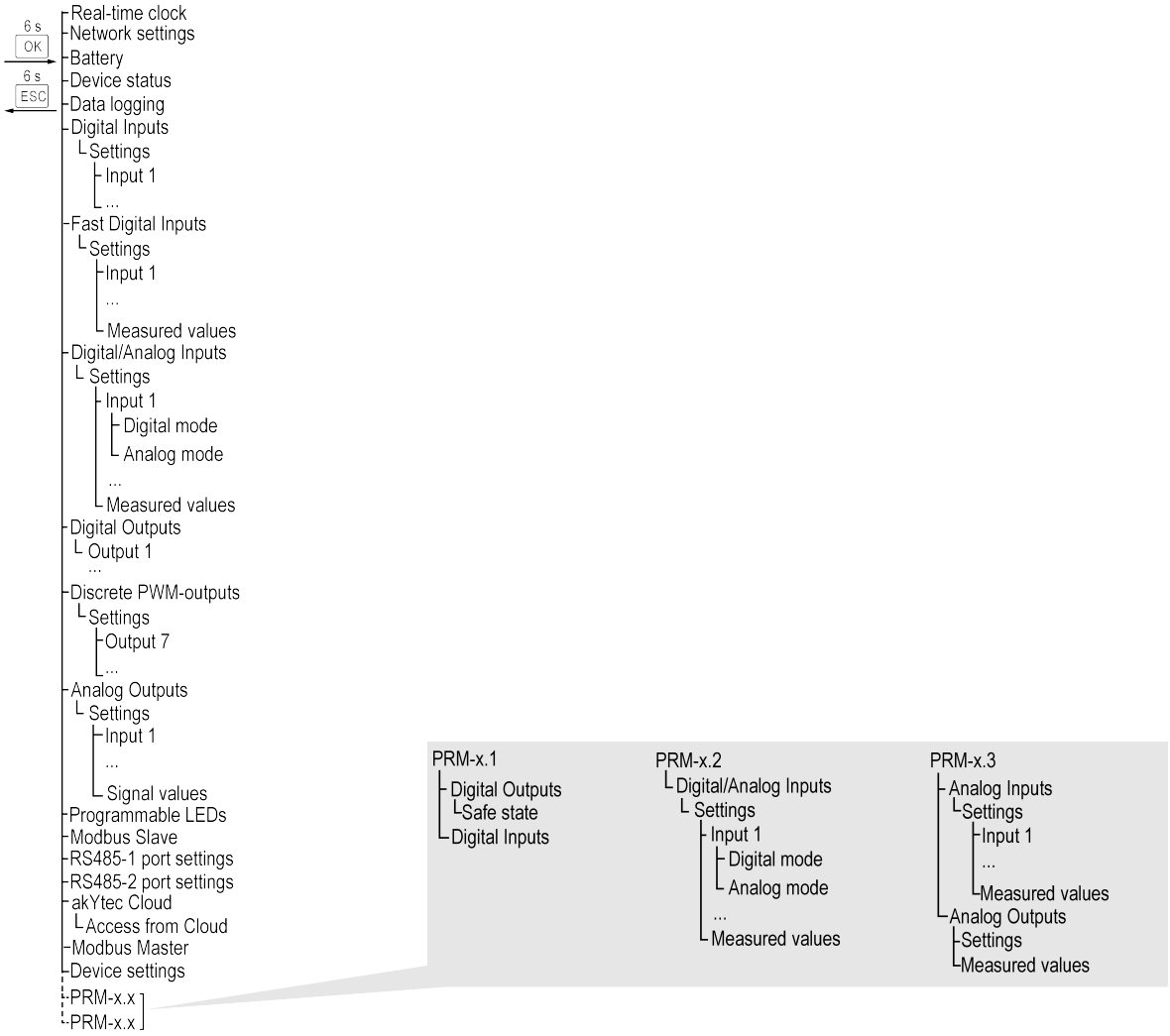



*Fig. 7.3 Menu structure*

*Table 7.9 System menu items*

| Item | Description |
|---|---|
| Program | Cycle time values and user program status are displayed |
| Digital inputs | The parameters are similar to those described in *Section 4.2.2* |
| Fast digital inputs | The parameters are similar to those described in *Section 4.2.3* |
| Digital outputs | The parameters are similar to those described in *Section 4.3.1* |
| Digital/Analog inputs | The parameters are similar to those described in *Section 4.2.1* |
| Analog outputs | The parameters are similar to those described in *Section 4.3.2* |
| Programmable LEDs | The parameters are similar to those described in *Section 4.4* |
| Modbus Slave | The parameters are similar to those described in *Section 4.8.4* |
| Modbus Master | The node contains the device polling parameters, similar to those configured in ALP |
| akYtec Cloud | Settings and permissions for akYtec Cloud (see *Section 4.11*) |
| RS485 port settings 1 (2) | The parameters are similar to those described in *Section 4.8.2.2* |

| Item | Description |
|---|---|
| Data logging | The parameters are similar to those described in *Section 4.5* |
| Network settings | Contains the Ethernet settings (see *Section 4.8.2.1*) |
| Real-time clock | The parameters are similar to those described in *Section 4.12* |
| Battery | The parameters are similar to those described in *Section 4.9* |
| Device status | The parameters are similar to those described in *Section 4.9* |
| Device settings | The parameters are similar to those described in *Section 4.10* |
| Password | The parameters are similar to those described in *Section 4.7.1* |
| PRM-x.x | These nodes contain parameters and statuses of the extension modules, if they are configured in ALP.<br>If the modules are not connected, the nodes are hidden. |

**NOTE**
If there is a user program recorded in the device, branch **Program** is displayed in the settings tree with the only unchangeable parameter in it: **Cycle time, ms**.

### 7.5 Error panel

The Error panel displays the errors containing status register bits (see *Table 7.11*). To switch to the error panel, press buttons SEL + ESC.

### 7.6 Jumps between display forms

To navigate between display forms, a user can create “jumps” in akYtec ALP. Jumps are performed by pressing buttons and by changing a variable. For more information about jumps, see the akYtec ALP help.

**NOTE**
When assigning a jump condition to a function button, don't forget that the user function of the button has a higher priority than its system function.

***Example:***
Buttons ▲ and ▼ cannot be used to scroll the lines inside a display if they are used as the jump condition for it. Button SEL with the jump condition assigned to it does not allow a user to switch to edit mode.

### 7.7 Programmable buttons

Besides the main functions (see *Table 7.1*), the device buttons can have additional functions assigned to them. Additional functions are assigned with the help of *Button Editor* in akYtec ALP. User-defined functions are assigned to:

– short press (less than 2 seconds),
– long press (more than 2 seconds),
– hold.

**NOTE**
In akYtec ALP, you cannot set functions for short/long press and hold at the same time.

Functions assigned to a button:

– switch to a display,
– change the value of a variable.

### 7.8 Auto-formatting mode

The auto-formatting mode displays a variable with maximum accuracy depending on the number of reserved characters. The auto-formatting mode is applied only for real variables. The mode can be enabled by selecting **Auto** in the "Decimal digits" field in akYtec ALP.

### 7.9 Modes of operation

As soon as the device is powered (either by main power or over USB), it reads the set position of the RUN/STOP. Then device runs self-testing.
Power over USB is sufficient for programming the device.

**NOTE**
***When the device is powered over USB, the inputs, outputs and the remaining interfaces are disabled.***

Once the main power is applied to the power terminals 1 and 2, the device starts the user's program if it has been loaded into the device memory.
The device operation modes are presented in the diagram below.

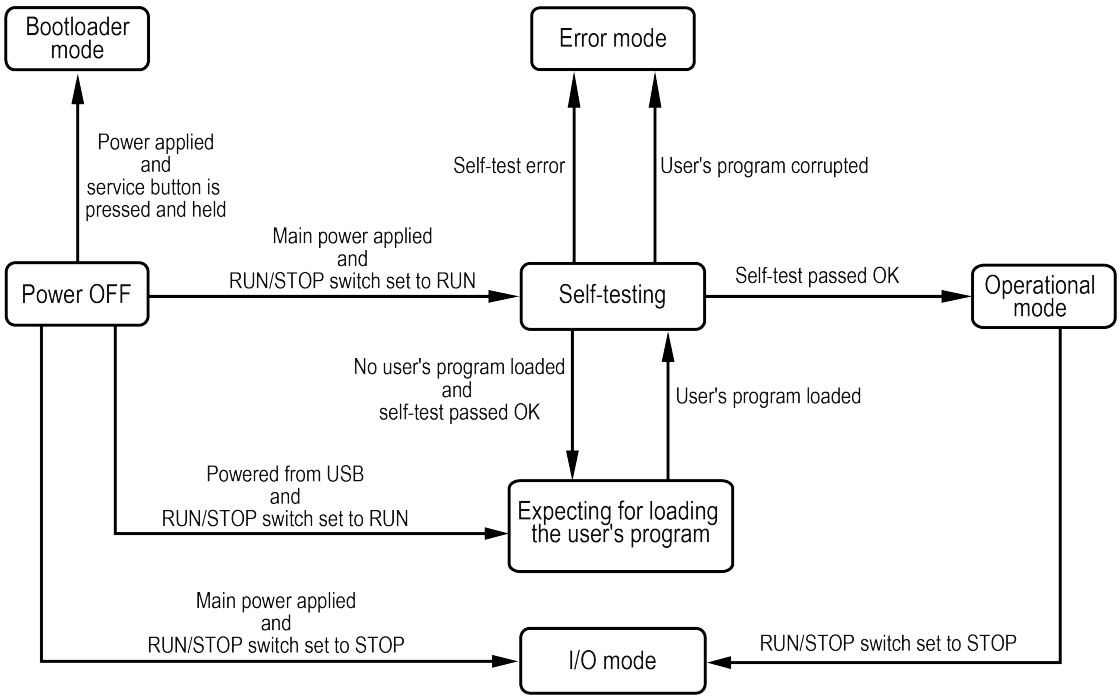



*Fig. 7.4 Operation modes diagram*

#### 7.9.1 Operational mode

When the device enters the operational mode, it runs the operational cycle which includes the sequence of the steps as follows:

1. Starting the operational cycle
2. Reading the input states
3. Executing the user's program code
4. Writing the output states
5. Return to the Step 1 (starting the operational cycle)

Once the operational cycle is started, the device reads the input states and then copies the read data into an input data memory area. Next, the device executes the user's program code using the copy of the input data available in the memory area.

### 7.9.2 Error mode

Device enters the error mode if any error occurred (see *Table 7.3* and *Table 7.4* for possible errors). Probable errors which cause device to enter error mode and related remedies are given in the table below.

*Table 7.10 Errors and error remedies*

| Indication ⚠ | Cause | Remedy |
|---|---|---|
| ON | Device failure | Contact to the service center |
| Flashing | Flashing together with ⚠ — RTC battery discharged | Replace the RTC battery (see *Section 8.2*) |

Type of an error occurred may be read from the Modbus status register if the Modbus connection is not failed and reading the status register is possible.
Status register bit assignment is given in the table below.

*Table 7.11 Bit assignment of the status register 61620 (0xF0B4)*

| Bit No. | Assignment |
|---|---|
| Bit 0 | Failure of digital inputs |
| Bit 1 | Failure of digital outputs |
| Bit 2 | Failure of analog inputs |
| Bit 3 | Failure of analog outputs |
| Bit 4 | Ethernet interface failure |
| Bit 5 | Not used |
| Bit 6 | USB interface failure |
| Bit 7 | Not used |
| Bit 8 | RS485 #1 failure |
| Bit 9 | RS485 #2 failure |
| Bit 10 | Not used |
| Bit 11 | RTC failure |
| Bit 12 | Power voltage is not available on power terminals 1 and 2 |
| Bit 13 | Firmware failure or cycle time exceeds 100 ms |
| Bit 14 | Not used |
| Bit 15 | Operation system error |
| Bit 16 | File system failure |
| Bit 17 | Built-in data storage has been formatted |
| Bit 18 | No operational parameters |
| Bit 19 | Firmware failure or firmware version does not match the user's program version |
| Bit 20 | No user's program available, factory reset has been implemented |
| Bit 21 | No archive available. Or archive recording failure. Or recording parameters to archive is disabled from ALP |
| Bit 22 | RUN/STOP switch is set to STOP position |
| Bit 23 | No user's program available |
| Bit 24 | Execution of the user's program is terminated |

| Bit No. | Assignment |
|---|---|
| Bit 25 | Not used |
| Bit 26 | Not used |
| Bit 27 | Not used |
| Bit 28 | Not used |
| Bit 29 | Not used |
| Bit 30 | Not used |
| Bit 31 | User's program failure in retain memory |

### 7.9.3 I/O mode

Once the RUN/STOP switch is set to STOP position, the user's program execution is terminated and the device enters I/O mode.
In I/O mode it is possible to read inputs and to control outputs, but there is no access to network variables.

### 7.9.4 Bootloader mode

**NOTE**
For the device in normal operation, firmware update should be performed according to *Section 7.14*.

When the device enters the bootloader mode, it is ready for updating its firmware over USB with no other device functionality supported.
The device enters the bootloader mode when:

– it is impossible to run the device firmware in the operational mode (see *Section 7.9.1*).
– the bootloader mode is forced by user.

The following steps must be implemented to force the bootloader mode:

1. Power down the device and disconnect the USB cable (if it was connected to the device earlier).
2. Press and hold the service button.
3. Apply power to the terminals 1 and 2 or connect USB cable to the device.

## 7.10 Extension modules

PRM extension modules are used to increase the number of I/O points. Please refer to the *section 6.6* for connection and installation of the extension modules. The operation of the extension modules is determined by user program in PR205.
The input polling time and the output state recording time of the extension modules are determined by complexity of the user program and they are the same as of PR205 inputs/outputs.

## 7.11 User program parameters

User variables can be assigned to the user program parameters in ALP.

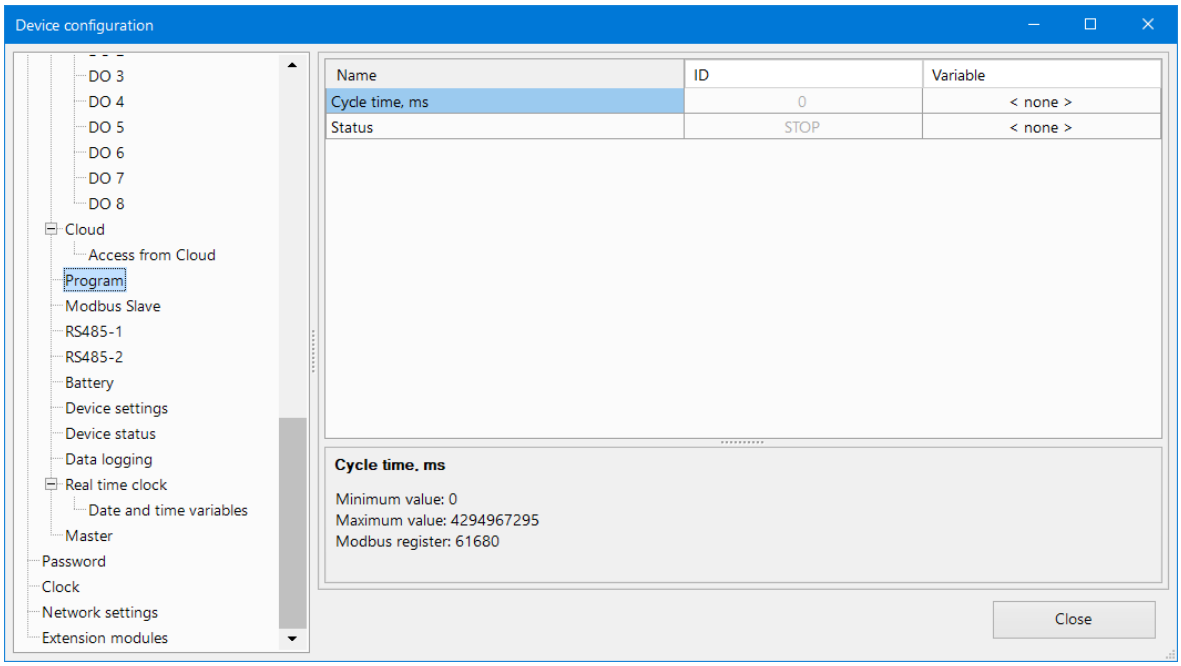



*Fig. 7.5 Assignment of variables to the user program parameters*

### 7.12 Real-time clock and retain memory

The device is equipped with the RTC and the retain memory.
The RTC is used for data logging. The retain memory is used for storing all settings of the device
The RTC and the retain memory are powered from the main supply voltage (applied to the terminals 1 and 2) when the device is powered on. Otherwise, the RTC and the retain memory are powered by the replaceable CR2032 battery.
A fully charged battery can supply RTC and retain memory continuously for 5 years. At the temperatures near the limits of the operating range (see *table 3.11*), the operating time of the RTC and the retain memory is reduced.
Please refer to *section 7.9.2* for RTC and retain memory related errors.
At the discharged battery, the following settings will be reset when the main supply voltage is removed from the device:

– RTC settings.
– the device settings which have been changed over Modbus or by the user program during device operation with the battery discharged.

If the discharged battery causes the reset of the settings, replace the battery. Please refer to *section* for the battery replacement.
The battery level can be checked in ALP.

### 7.13 Setting Date and Time from the front panel

To set date and time using buttons on the front panel:

1. Press and hold button OK. The system menu will appear on the screen.
2. Select **Real-time clock** by pressing button OK.
3. Press button SEL.
4. The cursor will start flashing in the **Time and Date (UTC)** parameter at seconds. You can use buttons ⌃ and ⌄ to increase or decrease the value. To switch to the next level, press button «. To save the changed values, press OK.

5. To change the Time zone parameter, use buttons  and  to select the required item and confirm the value by pressing OK.

To exit the system menu, press and hold button ESC.

### 7.14 Firmware update

**NOTE**
***To update the firmware, disconnect the device from akYtec Cloud.***

**NOTE**
***The user program is erased from the device on updating the device firmware with ALP.***

The firmware update is carried out with the programming and configuration interfaces.
Before starting the firmware update, perform the preparation steps as follows:

1. Prepare the PC with Windows 7 (SP1+)/8.1/10/11, akYtec ToolPro or ALP installed, and connect the PC to the Internet.
2. Install the device USB driver on the PC.

#### Firmware update in ALP

**NOTE**
***During the user program transfer to the device in ALP, the firmware is updated automatically***.

If the firmware update during user program transfer was unsuccessful, then **the forced firmware update** can be implemented. **The forced firmware update** can be made if the device is not detected in ALP, but the device connection is correctly displayed in the **Windows Device Manager**. Follow the steps below for **the forced firmware update**:

1. Connect the device to the PC using a microUSB — USB cable or an Ethernet cable.
2. Then:
   – In the Windows Device Manager, check which COM port the device is using if it is connected over USB;
   – Check network settings in akYtec ToolPro if the device is connected over Ethernet.
3. In ALP, click the menu item ***Device > Port settings*** and enter the read COM port in the open dialog.
4. In ALP, select menu item ***Device > Firmware update***.
5. The currently connected device modification will be displayed and proposed to confirm the firmware update.

   **NOTE**
   ***If the device modification is not displayed or the displayed modification does not match the connected device, please contact technical support to resolve the problem***.

6. Confirm to start the firmware update procedure.

When the firmware update is in progress, the loss of communication between the device and PC will cause damage of the device firmware. As result, the device will fail to operate. If this happened, repeat the forced firmware update to resolve the problem. If the above actions do not help, switch over to the Bootloader mode (see *Section 7.9.4*) and repeat the forced firmware update.

# 8 Maintenance

## 8.1 Maintenance

The safety requirements (see *Section 1.4*) must be observed when the maintenance is carried out.

**WARNING**
***Cut off all power before maintenance.***

The maintenance includes:
– cleaning of the housing and terminal blocks from dust, dirt and derbis;
– checking the device fastening;
– checking the wiring (connecting wires, terminal connections, absence of mechanical damages).

**NOTICE**
***The device should be cleaned with a dry or slightly damp cloth only. No abrasives or solvent-containing cleaners may be used.***

## 8.2 Battery replacement

**NOTE**
***The supply voltage may remain on when replacing the battery. This will prevent resetting the real-time clock and the device settings stored in the retain memory.***

Follow the steps below to replace the RTC battery:

1. Open the cover with the front panel.
2. 

   **NOTICE**
   ***Remove any static charge from your hands and the used tools before replacement of the battery.***

   Pick up the battery on the right side with a screwdriver and, while holding it, pull the battery out of its holder.
3. Insert a new battery observing polarity.
4. Close the cover.

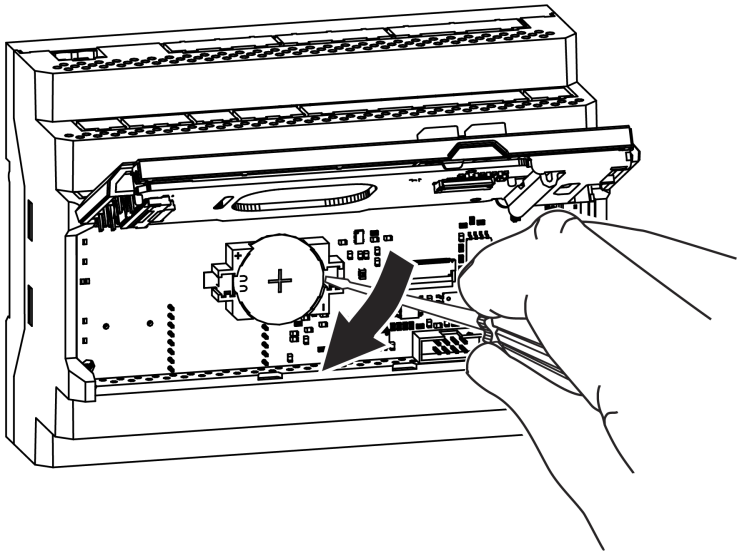


*Fig. 8.1 Battery replacement*

# 9 Transportation and storage

Pack the device in such a way as to protect it reliably against impact for storage and transportation. The original packaging provides optimum protection.
If the device is not taken immediately after delivery into operation, it must be carefully stored at a protected location. The device should not be stored in an atmosphere with chemically active substances.
The environmental conditions must be taken into account during transportation and storage.

**NOTICE**
***The device may have been damaged during transportation.***
***Check the device for transport damage and completeness!***
***Report the transport damage immediately to the shipper and akYtec GmbH!***

## 10 Scope of delivery

| | |
|---|---|
| PR205 | 1 |
| Short guide | 1 |
| Terminal blocks (set) | 1 |

## Appendix A. Terminal block layout

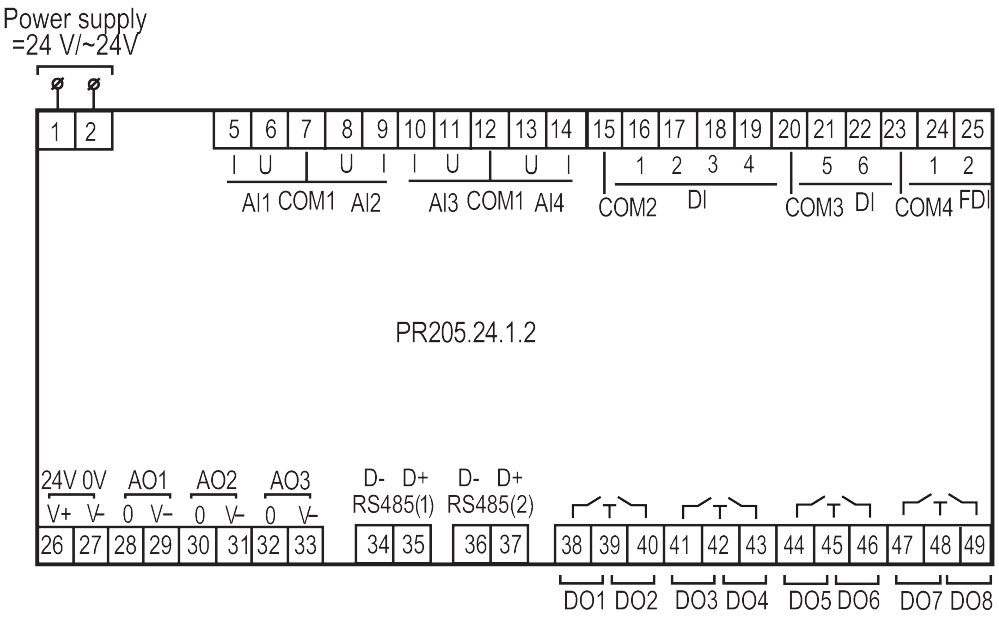



*Fig. A.1 PR205.24.1.2 terminal block layout*

*Table A.1 PR205.24.1.2 terminal assignment*

| No. | Marking | Description | No. | Marking | Description |
|---|---|---|---|---|---|
| 1 | 24 V | ”-” | 36 | D- | RS485 interface 2 |
| 2 | 24 V | ”+” | 37 | D+ | RS485 interface 2 |
| 15 | COM2 | DI1…DI4 common contact | 5 | I | AI1 current input |
| 16 | DI1 | DI1 digital input (24 V) | 6 | U | AI1 voltage input |
| 17 | DI2 | DI2 digital input (24 V) | 7 | COM1 | AI1…AI2 common contact |
| 18 | DI3 | DI3 digital input (24 V) | 8 | U | AI2 voltage input |
| 19 | DI4 | DI4 digital input (24 V) | 9 | I | AI2 current input |
| 20 | COM3 | DI5…DI6 common contact | 10 | I | AI3 current input |
| 21 | DI5 | DI5 digital input (24 V) | 11 | U | AI3 voltage input |
| 22 | DI6 | DI6 digital input (24 V) | 12 | COM1 | AI3…AI4 common contact |
| 23 | COM4 | FDI1…FDI2 common contact | 13 | U | AI4 voltage input |
| 24 | FDI1 | FDI1 fast digital input (24 V) | 14 | I | AI4 current input |
| 25 | FDI2 | FDI2 fast digital input (24 V) | 38 | DO1 | DO1 digital output |
| 26 | V+ | AO1…AO3 +24 VDC | 39 | T | DO1…DO2 common contact |
| 27 | V- | AO1…AO3 -24 VDC | 40 | DO2 | FDI4 digital input |
| 28 | 0 | AO1 | 41 | DO3 | RS485 interface 2 |
| 29 | V- | AO1 -24 VDC | 42 | T | DO3…DO4 common contact |
| 30 | 0 | AO2 | 43 | DO4 | DO4 digital output |
| 31 | V- | AO2 -24 VDC | 44 | DO5 | DO5 digital output |
| 32 | 0 | AO3 | 45 | T | DO5…DO6 common contact |
| 33 | V- | AO3 -24 VDC | 46 | DO6 | DO6 digital output |
| 34 | D- | RS485 interface 1 | 47 | DO7 | DO7 digital output |
| 35 | D+ | RS485 interface 1 | 48 | T | DO7…DO8 common contact |
| — | — | — | 49 | DO8 | DO8 digital output |

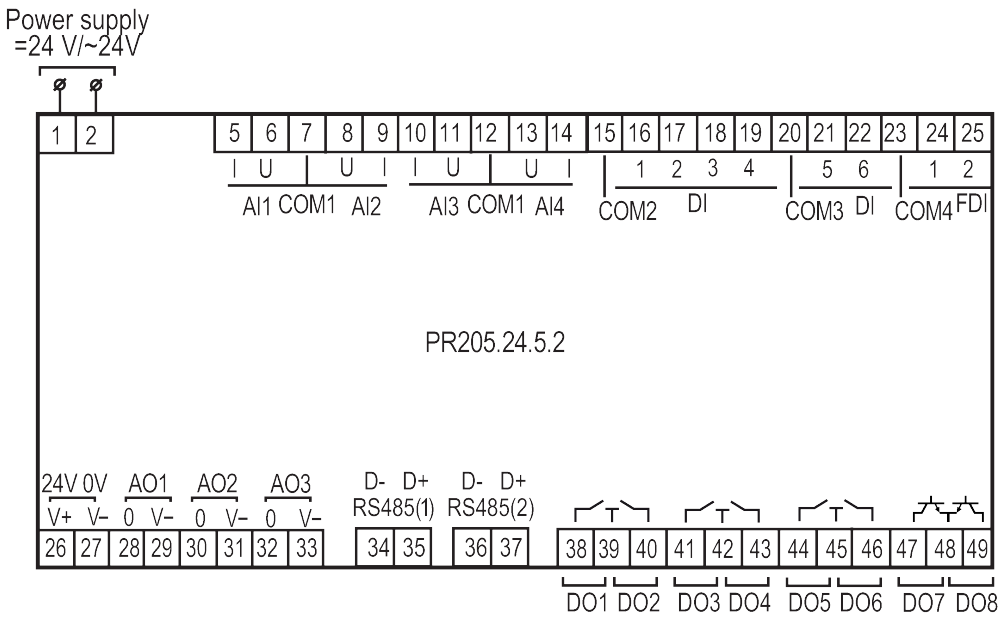



*Fig. A.2 PR205.24.5.2 terminal block layout*

**NOTE**
Outputs DO1...DO6 are of relay type.
Outputs DO7...DO8 are of NPN transistor type.

*Table A.2 205.24.5.2 terminal assignment*

| No. | Marking | Description | No. | Marking | Description |
|---|---|---|---|---|---|
| 1 | 24 V | ”-” | 36 | D- | RS485 interface 2 |
| 2 | 24 V | ”+” | 37 | D+ | RS485 interface 2 |
| 15 | COM2 | DI1…DI4 common contact | 5 | I | AI1 current input |
| 16 | DI1 | DI1 digital input (24 V) | 6 | U | AI1 voltage input |
| 17 | DI2 | DI2 digital input (24 V) | 7 | COM1 | AI1…AI2 common contact |
| 18 | DI3 | DI3 digital input (24 V) | 8 | U | AI2 voltage input |
| 19 | DI4 | DI4 digital input (24 V) | 9 | I | AI2 current input |
| 20 | COM3 | DI5…DI6 common contact | 10 | I | AI3 current input |
| 21 | DI5 | DI5 digital input (24 V) | 11 | U | AI3 voltage input |
| 22 | DI6 | DI6 digital input (24 V) | 12 | COM1 | AI3…AI4 common contact |
| 23 | COM4 | FDI1…FDI2 common contact | 13 | U | AI4 voltage input |
| 24 | FDI1 | FDI1 fast digital input (24 V) | 14 | I | AI4 current input |
| 25 | FDI2 | FDI2 fast digital input (24 V) | 38 | DO1 | DO1 digital output |
| 26 | V+ | AO1…AO3 +24 VDC | 39 | T | DO1…DO2 common contact |
| 27 | V- | AO1…AO3 -24 VDC | 40 | DO2 | FDI4 digital input |
| 28 | 0 | AO1 | 41 | DO3 | RS485 interface 2 |
| 29 | V- | AO1 -24 VDC | 42 | T | DO3…DO4 common contact |
| 30 | 0 | AO2 | 43 | DO4 | DO4 digital output |
| 31 | V- | AO2 -24 VDC | 44 | DO5 | DO5 digital output |
| 32 | 0 | AO3 | 45 | T | DO5…DO6 common contact |
| 33 | V- | AO3 -24 VDC | 46 | DO6 | DO6 digital output |
| 34 | D- | RS485 interface 1 | 47 | DO7 | DO7 transistor output 1 (collector) |

| No. | Mark-ing | Description | No. | Mark-ing | Description |
|---|---|---|---|---|---|
| 35 | D+ | RS485 interface 1 | 48 | T | Transistor outputs 1 and 2 (common emitter) |
| — | — | — | 49 | DO8 | DO8 transistor output 2 (collector) |

## Appendix B. Modbus register map

The addresses of Modbus registers, depending on the device modifications, are given in the tables below.

*Table B.1 PR205.24.1.2 Modbus registers*

| Parame-ter | Group | Address (dec) | Address (hex) | Number of registers | Read func-tion | Write func-tion | Data type |
|---|---|---|---|---|---|---|---|
| Time (ms) | Real time clock | 61563 | 0xF07B | 2 | 3 | - | Unsigned 32 |
| Time (UTC) | Real time clock | 61553 | 0xF071 | 2 | 3 | 16 | Date time 32 |
| Time zone | Real time clock | 61555 | 0xF073 | 1 | 3 | 16 | Enum 38 |
| Seconds | Date and time variables | 61557 | 0xF075 | 1 | 3 | - | Unsigned 8 |
| Minutes | Date and time variables | 61558 | 0xF076 | 1 | 3 | - | Unsigned 8 |
| Hours | Date and time variables | 61559 | 0xF077 | 1 | 3 | - | Unsigned 8 |
| Days | Date and time variables | 61560 | 0xF078 | 1 | 3 | - | Unsigned 8 |
| Months | Date and time variables | 61561 | 0xF079 | 1 | 3 | - | Unsigned 8 |
| Years | Date and time variables | 61562 | 0xF07A | 1 | 3 | - | Unsigned 16 |
| Day of the week | Date and time variables | 61556 | 0xF074 | 1 | 3 | - | Enum 8 |
| MAC address | Network | 61712 | 0xF110 | 9 | 3 | - | String 144 |
| IP address | Network | 20 | 0x0014 | 2 | 3 | 16 | Unsigned 32 |
| Subnet mask | Network | 22 | 0x0016 | 2 | 3 | 16 | Unsigned 32 |
| Gateway | Network | 24 | 0x0018 | 2 | 3 | 16 | Unsigned 32 |
| DNS server 1 | Network | 12 | 0x000C | 2 | 3 | 16 | Unsigned 32 |
| DNS server 2 | Network | 14 | 0x000E | 2 | 3 | 16 | Unsigned 32 |
| DHCP | Network | 32 | 0x0020 | 1 | 3 | 16 | Enum 2 |
| Apply now | Network | 33 | 0x0021 | 1 | 3 | 16 | Enum 2 |

| Parameter | Group | Address (dec) | Address (hex) | Number of registers | Read function | Write function | Data type |
|---|---|---|---|---|---|---|---|
| Connection status | Network | 34 | 0x0022 | 1 | 3 | - | Enum 4 |
| Voltage | Battery | 801 | 0x0321 | 1 | 3 | - | Unsigned 16 |
| Low limit | Battery | 800 | 0x0320 | 1 | 3 | - | Unsigned 16 |
| Status | Battery | 802 | 0x0322 | 1 | 3 | - | Enum 2 |
| Update period | Device status | 61624 | 0xF0B8 | 1 | 3 | 16 | Unsigned 8 |
| Status | Device status | 61620 | 0xF0B4 | 2 | 3 | - | Unsigned 32 |
| Extension 1. Module name | Device status | 6000 | 0x1770 | 8 | 3 | - | String 128 |
| Extension 1. Module FW version | Device status | 6016 | 0x1780 | 4 | 3 | - | String 64 |
| Extension 2. Module name | Device status | 6032 | 0x1790 | 8 | 3 | - | String 128 |
| Extension 2. Module FW version | Device status | 6048 | 0x17A0 | 4 | 3 | - | String 64 |
| Errors | Device status | 61626 | 0xF0BA | 1 | 3 | - | Unsigned 16 |
| Warnings | Device status | 61627 | 0xF0BB | 1 | 3 | - | Unsigned 16 |
| Logging interval | Data logging | 900 | 0x0384 | 1 | 3 | 16 | Unsigned 16 |
| Number of files | Data logging | 901 | 0x0385 | 1 | 3 | 16 | Unsigned 16 |
| File size | Data logging | 902 | 0x0386 | 1 | 3 | 16 | Unsigned 16 |
| Last log file ID | Data logging | 903 | 0x0387 | 1 | 3 | - | Unsigned 16 |
| Input bitmask | Digital inputs | 51 | 0x0033 | 1 | 3 | - | Unsigned 8 |
| Inversion bitmask | Digital inputs | 57 | 0x0039 | 1 | 3 | 16 | Unsigned 8 |
| De-bounce filter | DI 1 | 96 | 0x0060 | 1 | 3 | 16 | Unsigned 8 |

| Parameter | Group | Address (dec) | Address (hex) | Number of registers | Read function | Write function | Data type |
|---|---|---|---|---|---|---|---|
| Debounce filter | DI 2 | 97 | 0x0061 | 1 | 3 | 16 | Unsigned 8 |
| Debounce filter | DI 3 | 98 | 0x0062 | 1 | 3 | 16 | Unsigned 8 |
| Debounce filter | DI 4 | 99 | 0x0063 | 1 | 3 | 16 | Unsigned 8 |
| Debounce filter | DI 5 | 100 | 0x0064 | 1 | 3 | 16 | Unsigned 8 |
| Debounce filter | DI 6 | 101 | 0x0065 | 1 | 3 | 16 | Unsigned 8 |
| Input bitmask | Fast digital inputs | 52 | 0x0034 | 1 | 3 | - | Unsigned 8 |
| Inversion bitmask | Fast digital inputs | 58 | 0x003A | 1 | 3 | 16 | Unsigned 8 |
| Input mode | FDI 1 | 64 | 0x0040 | 1 | 3 | 16 | Enum 3 |
| Debounce filter | FDI 1 | 104 | 0x0068 | 1 | 3 | 16 | Unsigned 8 |
| Pulse edge | FDI 1 | 80 | 0x0050 | 1 | 3 | 16 | Enum 2 |
| Counter reset | FDI 1 | 224 | 0x00E0 | 1 | 3 | 16 | Enum 2 |
| Counter status | FDI 1 | 256 | 0x0100 | 1 | 3 | - | Enum 2 |
| Input mode | FDI 2 | 65 | 0x0041 | 1 | 3 | 16 | Enum 2 |
| Debounce filter | FDI 2 | 105 | 0x0069 | 1 | 3 | 16 | Unsigned 8 |
| Pulse edge | FDI 2 | 81 | 0x0051 | 1 | 3 | 16 | Enum 2 |
| Counter reset | FDI 2 | 225 | 0x00E1 | 1 | 3 | 16 | Enum 2 |
| Counter status | FDI 2 | 257 | 0x0101 | 1 | 3 | - | Enum 2 |
| FDI 1 | Measured values | 160 | 0x00A0 | 2 | 3 | - | Unsigned 32 |
| FDI 2 | Measured values | 162 | 0x00A2 | 2 | 3 | - | Unsigned 32 |
| Input bitmask | Analog inputs | 4000 | 0x0FA0 | 1 | 3 | - | Unsigned 8 |

| Parameter | Group | Address (dec) | Address (hex) | Number of registers | Read function | Write function | Data type |
|---|---|---|---|---|---|---|---|
| Inversion bitmask | Analog inputs | 4357 | 0x1105 | 1 | 3 | 16 | Unsigned 8 |
| Input mode | AI 1 | 4100 | 0x1004 | 1 | 3 | 16 | Enum 2 |
| De-bounce filter | Digital mode | 4108 | 0x100C | 1 | 3 | 16 | Unsigned 8 |
| LOW | Digital mode | 4111 | 0x100F | 2 | 3 | 16 | Float 32 |
| HIGH | Digital mode | 4109 | 0x100D | 2 | 3 | 16 | Float 32 |
| Input signal | Analog mode | 4101 | 0x1005 | 1 | 3 | 16 | Enum 28 |
| Analog filter | Analog mode | 4106 | 0x100A | 2 | 3 | 16 | Float 32 |
| Lower measuring limit | Analog mode | 4104 | 0x1008 | 2 | 3 | 16 | Float 32 |
| Upper measuring limit | Analog mode | 4102 | 0x1006 | 2 | 3 | 16 | Float 32 |
| Input mode | AI 2 | 4116 | 0x1014 | 1 | 3 | 16 | Enum 2 |
| De-bounce filter | Digital mode | 4124 | 0x101C | 1 | 3 | 16 | Unsigned 8 |
| LOW | Digital mode | 4127 | 0x101F | 2 | 3 | 16 | Float 32 |
| HIGH | Digital mode | 4125 | 0x101D | 2 | 3 | 16 | Float 32 |
| Input signal | Analog mode | 4117 | 0x1015 | 1 | 3 | 16 | Enum 28 |
| Analog filter | Analog mode | 4122 | 0x101A | 2 | 3 | 16 | Float 32 |
| Lower measuring limit | Analog mode | 4120 | 0x1018 | 2 | 3 | 16 | Float 32 |
| Upper measuring limit | Analog mode | 4118 | 0x1016 | 2 | 3 | 16 | Float 32 |
| Input mode | AI 3 | 4132 | 0x1024 | 1 | 3 | 16 | Enum 2 |
| De-bounce filter | Digital mode | 4140 | 0x102C | 1 | 3 | 16 | Unsigned 8 |
| LOW | Digital mode | 4143 | 0x102F | 2 | 3 | 16 | Float 32 |
| HIGH | Digital mode | 4141 | 0x102D | 2 | 3 | 16 | Float 32 |
| Input signal | Analog mode | 4133 | 0x1025 | 1 | 3 | 16 | Enum 28 |

| Parameter | Group | Address (dec) | Address (hex) | Number of registers | Read function | Write function | Data type |
|---|---|---|---|---|---|---|---|
| Analog filter | Analog mode | 4138 | 0x102A | 2 | 3 | 16 | Float 32 |
| Lower measuring limit | Analog mode | 4136 | 0x1028 | 2 | 3 | 16 | Float 32 |
| Upper measuring limit | Analog mode | 4134 | 0x1026 | 2 | 3 | 16 | Float 32 |
| Input mode | AI 4 | 4148 | 0x1034 | 1 | 3 | 16 | Enum 2 |
| Debounce filter | Digital mode | 4156 | 0x103C | 1 | 3 | 16 | Unsigned 8 |
| LOW | Digital mode | 4159 | 0x103F | 2 | 3 | 16 | Float 32 |
| HIGH | Digital mode | 4157 | 0x103D | 2 | 3 | 16 | Float 32 |
| Input signal | Analog mode | 4149 | 0x1035 | 1 | 3 | 16 | Enum 28 |
| Analog filter | Analog mode | 4154 | 0x103A | 2 | 3 | 16 | Float 32 |
| Lower measuring limit | Analog mode | 4152 | 0x1038 | 2 | 3 | 16 | Float 32 |
| Upper measuring limit | Analog mode | 4150 | 0x1036 | 2 | 3 | 16 | Float 32 |
| AI 1 | Measured values | 4002 | 0x0FA2 | 2 | 3 | - | Float 32 |
| AI 2 | Measured values | 4004 | 0x0FA4 | 2 | 3 | - | Float 32 |
| AI 3 | Measured values | 4006 | 0x0FA6 | 2 | 3 | - | Float 32 |
| AI 4 | Measured values | 4008 | 0x0FA8 | 2 | 3 | - | Float 32 |
| AI 1 | Input statuses | 4014 | 0x0FAE | 1 | 3 | - | Enum 11 |
| AI 2 | Input statuses | 4015 | 0x0FAF | 1 | 3 | - | Enum 11 |
| AI 3 | Input statuses | 4016 | 0x0FB0 | 1 | 3 | - | Enum 11 |
| AI 4 | Input statuses | 4017 | 0x0FB1 | 1 | 3 | - | Enum 11 |
| New output bitmask | Digital outputs | 470 | 0x01D6 | 1 | 3 | 16 | Unsigned 8 |
| Output bitmask | Digital outputs | 468 | 0x01D4 | 1 | 3 | - | Unsigned 8 |

| Parame-ter | Group | Address (dec) | Address (hex) | Number of registers | Read func-tion | Write func-tion | Data type |
|---|---|---|---|---|---|---|---|
| Safe state | DO 1 | 474 | 0x01DA | 1 | 3 | 16 | Enum 3 |
| Safe state | DO 2 | 475 | 0x01DB | 1 | 3 | 16 | Enum 3 |
| Safe state | DO 3 | 476 | 0x01DC | 1 | 3 | 16 | Enum 3 |
| Safe state | DO 4 | 477 | 0x01DD | 1 | 3 | 16 | Enum 3 |
| Safe state | DO 5 | 478 | 0x01DE | 1 | 3 | 16 | Enum 3 |
| Safe state | DO 6 | 479 | 0x01DF | 1 | 3 | 16 | Enum 3 |
| Safe state | DO 7 | 480 | 0x01E0 | 1 | 3 | 16 | Enum 3 |
| Safe state | DO 8 | 481 | 0x01E1 | 1 | 3 | 16 | Enum 3 |
| Output mode | AO 1 | 3160 | 0x0C58 | 1 | 3 | 16 | Enum 3 |
| Output status | AO 1 | 3128 | 0x0C38 | 1 | 3 | - | Enum 6 |
| Safe state | AO 1 | 3032 | 0x0BD8 | 2 | 3 | 16 | Float 32 |
| Output mode | AO 2 | 3161 | 0x0C59 | 1 | 3 | 16 | Enum 3 |
| Output status | AO 2 | 3129 | 0x0C39 | 1 | 3 | - | Enum 6 |
| Safe state | AO 2 | 3034 | 0x0BDA | 2 | 3 | 16 | Float 32 |
| Output mode | AO 3 | 3162 | 0x0C5A | 1 | 3 | 16 | Enum 3 |
| Output status | AO 3 | 3130 | 0x0C3A | 1 | 3 | - | Enum 6 |
| Safe state | AO 3 | 3036 | 0x0BDC | 2 | 3 | 16 | Float 32 |
| AO 1 | Output signal | 3000 | 0x0BB8 | 2 | 3 | 16 | Float 32 |
| AO 2 | Output signal | 3002 | 0x0BBA | 2 | 3 | 16 | Float 32 |
| AO 3 | Output signal | 3004 | 0x0BBC | 2 | 3 | 16 | Float 32 |
| LED bitmask | Programma-ble LEDs (Fn) | 601 | 0x0259 | 1 | 3 | - | Unsigned 8 |
| New LED bitmask | Programma-ble LEDs (Fn) | 600 | 0x0258 | 1 | 3 | 16 | Unsigned 8 |

| Parameter | Group | Address (dec) | Address (hex) | Number of registers | Read function | Write function | Data type |
|---|---|---|---|---|---|---|---|
| Safe state timeout | Modbus Slave | 700 | 0x02BC | 1 | 3 | 16 | Unsigned 8 |
| Baudrate | RS485-1 | 750 | 0x02EE | 1 | 3 | 16 | Enum 6 |
| Data bits | RS485-1 | 751 | 0x02EF | 1 | 3 | 16 | Enum 2 |
| Parity | RS485-1 | 752 | 0x02F0 | 1 | 3 | 16 | Enum 3 |
| Stop bits | RS485-1 | 753 | 0x02F1 | 1 | 3 | 16 | Enum 2 |
| Slave address | RS485-1 | 754 | 0x02F2 | 1 | 3 | 16 | Unsigned 8 |
| Baudrate | RS485-2 | 760 | 0x02F8 | 1 | 3 | 16 | Enum 6 |
| Data bits | RS485-2 | 761 | 0x02F9 | 1 | 3 | 16 | Enum 2 |
| Parity | RS485-2 | 762 | 0x02FA | 1 | 3 | 16 | Enum 3 |
| Stop bits | RS485-2 | 763 | 0x02FB | 1 | 3 | 16 | Enum 2 |
| Slave address | RS485-2 | 764 | 0x02FC | 1 | 3 | 16 | Unsigned 8 |
| Cloud connection | Cloud | 35 | 0x0023 | 1 | 3 | 16 | Enum 2 |
| Connection status | Cloud | 36 | 0x0024 | 1 | 3 | - | Enum 5 |
| Configuration | Access from Cloud | 701 | 0x02BD | 1 | 3 | 16 | Enum 2 |
| Output control | Access from Cloud | 702 | 0x02BE | 1 | 3 | 16 | Enum 2 |
| Modbus registers access | Access from Cloud | 703 | 0x02BF | 1 | 3 | 16 | Enum 4 |
| Availability of exchange | Sharing with devices | 2008 | 0x07D8 | 2 | 3 | - | Unsigned 32 |
| Enable sharing | Sharing with devices | 2010 | 0x07DA | 2 | 3 | 16 | Unsigned 32 |
| Cycle time | Program | 61680 | 0xF0F0 | 2 | 3 | - | Unsigned 32 |
| Status | Program | 61682 | 0xF0F2 | 1 | 3 | - | Enum 2 |
| Screen backlight | Device settings | 768 | 0x0300 | 1 | 3 | 16 | Enum 5 |

*Table B.2 PR205.24.5.2 Modbus registers*

| Parameter | Group | Address (dec) | Address (hex) | Number of registers | Read function | Write function | Data type |
|---|---|---|---|---|---|---|---|
| Time (ms) | Real time clock | 61563 | 0xF07B | 2 | 3 | - | Unsigned 32 |
| Time (UTC) | Real time clock | 61553 | 0xF071 | 2 | 3 | 16 | Date time 32 |
| Time zone | Real time clock | 61555 | 0xF073 | 1 | 3 | 16 | Enum 38 |
| Seconds | Date and time variables | 61557 | 0xF075 | 1 | 3 | - | Unsigned 8 |
| Minutes | Date and time variables | 61558 | 0xF076 | 1 | 3 | - | Unsigned 8 |
| Hours | Date and time variables | 61559 | 0xF077 | 1 | 3 | - | Unsigned 8 |
| Days | Date and time variables | 61560 | 0xF078 | 1 | 3 | - | Unsigned 8 |
| Months | Date and time variables | 61561 | 0xF079 | 1 | 3 | - | Unsigned 8 |
| Years | Date and time variables | 61562 | 0xF07A | 1 | 3 | - | Unsigned 16 |
| Day of the week | Date and time variables | 61556 | 0xF074 | 1 | 3 | - | Enum 8 |
| MAC address | Network | 61712 | 0xF110 | 9 | 3 | - | String 144 |
| IP address | Network | 20 | 0x0014 | 2 | 3 | 16 | Unsigned 32 |
| Subnet mask | Network | 22 | 0x0016 | 2 | 3 | 16 | Unsigned 32 |
| Gateway | Network | 24 | 0x0018 | 2 | 3 | 16 | Unsigned 32 |
| DNS server 1 | Network | 12 | 0x000C | 2 | 3 | 16 | Unsigned 32 |
| DNS server 2 | Network | 14 | 0x000E | 2 | 3 | 16 | Unsigned 32 |
| DHCP | Network | 32 | 0x0020 | 1 | 3 | 16 | Enum 2 |
| Apply now | Network | 33 | 0x0021 | 1 | 3 | 16 | Enum 2 |
| Connection status | Network | 34 | 0x0022 | 1 | 3 | - | Enum 4 |

| Parameter | Group | Address (dec) | Address (hex) | Number of registers | Read function | Write function | Data type |
|---|---|---|---|---|---|---|---|
| Voltage | Battery | 801 | 0x0321 | 1 | 3 | - | Unsigned 16 |
| Low limit | Battery | 800 | 0x0320 | 1 | 3 | - | Unsigned 16 |
| Status | Battery | 802 | 0x0322 | 1 | 3 | - | Enum 2 |
| Update period | Device status | 61624 | 0xF0B8 | 1 | 3 | 16 | Unsigned 8 |
| Status | Device status | 61620 | 0xF0B4 | 2 | 3 | - | Unsigned 32 |
| Extension 1. Module name | Device status | 6000 | 0x1770 | 8 | 3 | - | String 128 |
| Extension 1. Module FW version | Device status | 6016 | 0x1780 | 4 | 3 | - | String 64 |
| Extension 2. Module name | Device status | 6032 | 0x1790 | 8 | 3 | - | String 128 |
| Extension 2. Module FW version | Device status | 6048 | 0x17A0 | 4 | 3 | - | String 64 |
| Errors | Device status | 61626 | 0xF0BA | 1 | 3 | - | Unsigned 16 |
| Warnings | Device status | 61627 | 0xF0BB | 1 | 3 | - | Unsigned 16 |
| Logging interval | Data logging | 900 | 0x0384 | 1 | 3 | 16 | Unsigned 16 |
| Number of files | Data logging | 901 | 0x0385 | 1 | 3 | 16 | Unsigned 16 |
| File size | Data logging | 902 | 0x0386 | 1 | 3 | 16 | Unsigned 16 |
| Last log file ID | Data logging | 903 | 0x0387 | 1 | 3 | - | Unsigned 16 |
| Input bitmask | Digital inputs | 51 | 0x0033 | 1 | 3 | - | Unsigned 8 |

| Parameter | Group | Address (dec) | Address (hex) | Number of registers | Read function | Write function | Data type |
|---|---|---|---|---|---|---|---|
| Inversion bitmask | Digital inputs | 57 | 0x0039 | 1 | 3 | 16 | Unsigned 8 |
| Debounce filter | DI 1 | 96 | 0x0060 | 1 | 3 | 16 | Unsigned 8 |
| Debounce filter | DI 2 | 97 | 0x0061 | 1 | 3 | 16 | Unsigned 8 |
| Debounce filter | DI 3 | 98 | 0x0062 | 1 | 3 | 16 | Unsigned 8 |
| Debounce filter | DI 4 | 99 | 0x0063 | 1 | 3 | 16 | Unsigned 8 |
| Debounce filter | DI 5 | 100 | 0x0064 | 1 | 3 | 16 | Unsigned 8 |
| Debounce filter | DI 6 | 101 | 0x0065 | 1 | 3 | 16 | Unsigned 8 |
| Input bitmask | Fast digital inputs | 52 | 0x0034 | 1 | 3 | - | Unsigned 8 |
| Inversion bitmask | Fast digital inputs | 58 | 0x003A | 1 | 3 | 16 | Unsigned 8 |
| Input mode | FDI 1 | 64 | 0x0040 | 1 | 3 | 16 | Enum 3 |
| Debounce filter | FDI 1 | 104 | 0x0068 | 1 | 3 | 16 | Unsigned 8 |
| Pulse edge | FDI 1 | 80 | 0x0050 | 1 | 3 | 16 | Enum 2 |
| Counter reset | FDI 1 | 224 | 0x00E0 | 1 | 3 | 16 | Enum 2 |
| Counter status | FDI 1 | 256 | 0x0100 | 1 | 3 | - | Enum 2 |
| Input mode | FDI 2 | 65 | 0x0041 | 1 | 3 | 16 | Enum 2 |
| Debounce filter | FDI 2 | 105 | 0x0069 | 1 | 3 | 16 | Unsigned 8 |
| Pulse edge | FDI 2 | 81 | 0x0051 | 1 | 3 | 16 | Enum 2 |
| Counter reset | FDI 2 | 225 | 0x00E1 | 1 | 3 | 16 | Enum 2 |
| Counter status | FDI 2 | 257 | 0x0101 | 1 | 3 | - | Enum 2 |
| FDI 1 | Measured values | 160 | 0x00A0 | 2 | 3 | - | Unsigned 32 |
| FDI 2 | Measured values | 162 | 0x00A2 | 2 | 3 | - | Unsigned 32 |

| Parame-ter | Group | Address (dec) | Address (hex) | Number of registers | Read func-tion | Write func-tion | Data type |
|---|---|---|---|---|---|---|---|
| Input bitmask | Analog inputs | 4000 | 0x0FA0 | 1 | 3 | - | Un-signed 8 |
| Inversion bitmask | Analog inputs | 4357 | 0x1105 | 1 | 3 | 16 | Un-signed 8 |
| Input mode | AI 1 | 4100 | 0x1004 | 1 | 3 | 16 | Enum 2 |
| Debounce filter | Digital mode | 4108 | 0x100C | 1 | 3 | 16 | Un-signed 8 |
| LOW | Digital mode | 4111 | 0x100F | 2 | 3 | 16 | Float 32 |
| HIGH | Digital mode | 4109 | 0x100D | 2 | 3 | 16 | Float 32 |
| Input signal | Analog mode | 4101 | 0x1005 | 1 | 3 | 16 | Enum 28 |
| Analog filter | Analog mode | 4106 | 0x100A | 2 | 3 | 16 | Float 32 |
| Lower measur-ing limit | Analog mode | 4104 | 0x1008 | 2 | 3 | 16 | Float 32 |
| Upper measur-ing limit | Analog mode | 4102 | 0x1006 | 2 | 3 | 16 | Float 32 |
| Input mode | AI 2 | 4116 | 0x1014 | 1 | 3 | 16 | Enum 2 |
| Debounce filter | Digital mode | 4124 | 0x101C | 1 | 3 | 16 | Un-signed 8 |
| LOW | Digital mode | 4127 | 0x101F | 2 | 3 | 16 | Float 32 |
| HIGH | Digital mode | 4125 | 0x101D | 2 | 3 | 16 | Float 32 |
| Input signal | Analog mode | 4117 | 0x1015 | 1 | 3 | 16 | Enum 28 |
| Analog filter | Analog mode | 4122 | 0x101A | 2 | 3 | 16 | Float 32 |
| Lower measur-ing limit | Analog mode | 4120 | 0x1018 | 2 | 3 | 16 | Float 32 |
| Upper measur-ing limit | Analog mode | 4118 | 0x1016 | 2 | 3 | 16 | Float 32 |
| Input mode | AI 3 | 4132 | 0x1024 | 1 | 3 | 16 | Enum 2 |
| Debounce filter | Digital mode | 4140 | 0x102C | 1 | 3 | 16 | Un-signed 8 |
| LOW | Digital mode | 4143 | 0x102F | 2 | 3 | 16 | Float 32 |
| HIGH | Digital mode | 4141 | 0x102D | 2 | 3 | 16 | Float 32 |

| Parameter | Group | Address (dec) | Address (hex) | Number of registers | Read function | Write function | Data type |
|---|---|---|---|---|---|---|---|
| Input signal | Analog mode | 4133 | 0x1025 | 1 | 3 | 16 | Enum 28 |
| Analog filter | Analog mode | 4138 | 0x102A | 2 | 3 | 16 | Float 32 |
| Lower measuring limit | Analog mode | 4136 | 0x1028 | 2 | 3 | 16 | Float 32 |
| Upper measuring limit | Analog mode | 4134 | 0x1026 | 2 | 3 | 16 | Float 32 |
| Input mode | AI 4 | 4148 | 0x1034 | 1 | 3 | 16 | Enum 2 |
| Debounce filter | Digital mode | 4156 | 0x103C | 1 | 3 | 16 | Unsigned 8 |
| LOW | Digital mode | 4159 | 0x103F | 2 | 3 | 16 | Float 32 |
| HIGH | Digital mode | 4157 | 0x103D | 2 | 3 | 16 | Float 32 |
| Input signal | Analog mode | 4149 | 0x1035 | 1 | 3 | 16 | Enum 28 |
| Analog filter | Analog mode | 4154 | 0x103A | 2 | 3 | 16 | Float 32 |
| Lower measuring limit | Analog mode | 4152 | 0x1038 | 2 | 3 | 16 | Float 32 |
| Upper measuring limit | Analog mode | 4150 | 0x1036 | 2 | 3 | 16 | Float 32 |
| AI 1 | Measured values | 4002 | 0x0FA2 | 2 | 3 | - | Float 32 |
| AI 2 | Measured values | 4004 | 0x0FA4 | 2 | 3 | - | Float 32 |
| AI 3 | Measured values | 4006 | 0x0FA6 | 2 | 3 | - | Float 32 |
| AI 4 | Measured values | 4008 | 0x0FA8 | 2 | 3 | - | Float 32 |
| AI 1 | Input statuses | 4014 | 0x0FAE | 1 | 3 | - | Enum 11 |
| AI 2 | Input statuses | 4015 | 0x0FAF | 1 | 3 | - | Enum 11 |
| AI 3 | Input statuses | 4016 | 0x0FB0 | 1 | 3 | - | Enum 11 |
| AI 4 | Input statuses | 4017 | 0x0FB1 | 1 | 3 | - | Enum 11 |
| New output bitmask | Digital outputs | 470 | 0x01D6 | 1 | 3 | 16 | Unsigned 8 |

| Parame-ter | Group | Address (dec) | Address (hex) | Number of registers | Read func-tion | Write func-tion | Data type |
|---|---|---|---|---|---|---|---|
| Output bitmask | Digital outputs | 468 | 0x01D4 | 1 | 3 | - | Un-signed 8 |
| Safe state | DO 1 | 474 | 0x01DA | 1 | 3 | 16 | Enum 3 |
| Safe state | DO 2 | 475 | 0x01DB | 1 | 3 | 16 | Enum 3 |
| Safe state | DO 3 | 476 | 0x01DC | 1 | 3 | 16 | Enum 3 |
| Safe state | DO 4 | 477 | 0x01DD | 1 | 3 | 16 | Enum 3 |
| Safe state | DO 5 | 478 | 0x01DE | 1 | 3 | 16 | Enum 3 |
| Safe state | DO 6 | 479 | 0x01DF | 1 | 3 | 16 | Enum 3 |
| State | Discrete PWM-outputs | 306 | 0x0132 | 1 | 3 | - | Un-signed 8 |
| Operating mode | Output 7 | 272 | 0x0110 | 1 | 3 | 16 | Enum 2 |
| Period | Output 7 | 308 | 0x0134 | 1 | 3 | 16 | Un-signed 16 |
| Minimum pulse duration | Output 7 | 404 | 0x0194 | 1 | 3 | 16 | Un-signed 16 |
| Safe state | Output 7 | 538 | 0x021A | 2 | 3 | 16 | Float 32 |
| Operating mode | Output 8 | 273 | 0x0111 | 1 | 3 | 16 | Enum 2 |
| Period | Output 8 | 309 | 0x0135 | 1 | 3 | 16 | Un-signed 16 |
| Minimum pulse duration | Output 8 | 405 | 0x0195 | 1 | 3 | 16 | Un-signed 16 |
| Safe state | Output 8 | 540 | 0x021C | 2 | 3 | 16 | Float 32 |
| Output 7 | Duty cycle | 340 | 0x0154 | 2 | 3 | 16 | Float 32 |
| Output 8 | Duty cycle | 342 | 0x0156 | 2 | 3 | 16 | Float 32 |
| Output mode | AO 1 | 3160 | 0x0C58 | 1 | 3 | 16 | Enum 3 |
| Output status | AO 1 | 3128 | 0x0C38 | 1 | 3 | - | Enum 6 |
| Safe state | AO 1 | 3032 | 0x0BD8 | 2 | 3 | 16 | Float 32 |
| Output mode | AO 2 | 3161 | 0x0C59 | 1 | 3 | 16 | Enum 3 |
| Output status | AO 2 | 3129 | 0x0C39 | 1 | 3 | - | Enum 6 |
| Safe state | AO 2 | 3034 | 0x0BDA | 2 | 3 | 16 | Float 32 |
| Output mode | AO 3 | 3162 | 0x0C5A | 1 | 3 | 16 | Enum 3 |

| Parameter | Group | Address (dec) | Address (hex) | Number of registers | Read function | Write function | Data type |
|---|---|---|---|---|---|---|---|
| Output status | AO 3 | 3130 | 0x0C3A | 1 | 3 | - | Enum 6 |
| Safe state | AO 3 | 3036 | 0x0BDC | 2 | 3 | 16 | Float 32 |
| AO 1 | Output signal | 3000 | 0x0BB8 | 2 | 3 | 16 | Float 32 |
| AO 2 | Output signal | 3002 | 0x0BBA | 2 | 3 | 16 | Float 32 |
| AO 3 | Output signal | 3004 | 0x0BBC | 2 | 3 | 16 | Float 32 |
| LED bitmask | Programmable LEDs (Fn) | 601 | 0x0259 | 1 | 3 | - | Unsigned 8 |
| New LED bitmask | Programmable LEDs (Fn) | 600 | 0x0258 | 1 | 3 | 16 | Unsigned 8 |
| Safe state timeout | Modbus Slave | 700 | 0x02BC | 1 | 3 | 16 | Unsigned 8 |
| Baudrate | RS485-1 | 750 | 0x02EE | 1 | 3 | 16 | Enum 6 |
| Data bits | RS485-1 | 751 | 0x02EF | 1 | 3 | 16 | Enum 2 |
| Parity | RS485-1 | 752 | 0x02F0 | 1 | 3 | 16 | Enum 3 |
| Stop bits | RS485-1 | 753 | 0x02F1 | 1 | 3 | 16 | Enum 2 |
| Slave address | RS485-1 | 754 | 0x02F2 | 1 | 3 | 16 | Unsigned 8 |
| Baudrate | RS485-2 | 760 | 0x02F8 | 1 | 3 | 16 | Enum 6 |
| Data bits | RS485-2 | 761 | 0x02F9 | 1 | 3 | 16 | Enum 2 |
| Parity | RS485-2 | 762 | 0x02FA | 1 | 3 | 16 | Enum 3 |
| Stop bits | RS485-2 | 763 | 0x02FB | 1 | 3 | 16 | Enum 2 |
| Slave address | RS485-2 | 764 | 0x02FC | 1 | 3 | 16 | Unsigned 8 |
| Cloud connection | Cloud | 35 | 0x0023 | 1 | 3 | 16 | Enum 2 |
| Connection status | Cloud | 36 | 0x0024 | 1 | 3 | - | Enum 5 |
| Configuration | Access from Cloud | 701 | 0x02BD | 1 | 3 | 16 | Enum 2 |
| Output control | Access from Cloud | 702 | 0x02BE | 1 | 3 | 16 | Enum 2 |
| Modbus registers access | Access from Cloud | 703 | 0x02BF | 1 | 3 | 16 | Enum 4 |
| Availability of exchange | Sharing with devices | 2008 | 0x07D8 | 2 | 3 | - | Unsigned 32 |
| Enable sharing | Sharing with devices | 2010 | 0x07DA | 2 | 3 | 16 | Unsigned 32 |

| Parame-ter | Group | Address (dec) | Address (hex) | Number of registers | Read func-tion | Write func-tion | Data type |
|---|---|---|---|---|---|---|---|
| Cycle time | Program | 61680 | 0xF0F0 | 2 | 3 | - | Un-signed 32 |
| Status | Program | 61682 | 0xF0F2 | 1 | 3 | - | Enum 2 |
| Screen backlight | Device settings | 768 | 0x0300 | 1 | 3 | 16 | Enum 5 |

## Appendix C. Initialization vector for the archive file encryption

A cryptographic hash function should be used as an initialization vector for the archive file decryption. The hash function must return 8 bytes (LONG LONG type variable).
Please refer to the example of the hash function implemented in the C programming language given below.

```
typedef union {
        struct {
                unsigned long lo;
                unsigned long hi;
        };
        long long hilo;
}LONG_LONG;

long long Hash8(const char *str) { // based on Rot13
        LONG_LONG temp;
        temp.lo = 0;
        temp.hi = 0;
        for ( ; *str; )
        {
                temp.lo += (unsigned char) (*str);
                temp.lo –= (temp.lo << 13) | (temp.lo >> 19);
                str++;
                if (!str) break;
                temp.hi += (unsigned char) (*str);
                temp.hi –= (temp.hi << 13) | (temp.hi >> 19);
            str++;
        }
       return temp.hilo;
}
```